AFCRL-65-306

MAGNETIC TAPE COPIES OF MIT GEOPHYSICS PROGRAM SET II

(TIME SERIES PROGRAMS FOR THE IBM 709, 7090, 7094)

S. M. Simpson, Jr.

Massachusetts Institute of Technology
Cambridge 39, Massachusetts

Contract No. AF19(604)-7378

Project No. 8652

Task No. 865203

Scientific Report No. 10

March 31, 1965

Work Sponsored by Advanced Research Projects Agency

Project Vela-Uniform

ARPA Order No. 180-61, Amendment 2

Prepared
for

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS

CLEARINGHOUSE FOR FEDERAL SCIENTIFIC AND TECHNICAL INFORMATION, CFSTI
INPUT SECTION 410.11

LIMITATIONS IN REPRODUCTION QUALITY OF TECHNICAL ABSTRACT BULLETIN
DOCUMENTS, DEFENSE DOCUMENTATION CENTER (DDC)

AD 616254

☐ 1. AVAILABLE ONLY FOR REFERENCE USE AT DDC FIELD SERVICES.
COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☒ 2. AVAILABLE COPY WILL NOT PERMIT FULLY LEGIBLE REPRODUCTION.
REPRODUCTION WILL BE MADE IF REQUESTED BY USERS OF DDC.

☒ A. COPY IS AVAILABLE FOR PUBLIC SALE.

☐ B. COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☐ 3. LIMITED NUMBER OF COPIES CONTAINING COLOR OTHER THAN BLACK
AND WHITE ARE AVAILABLE UNTIL STOCK IS EXHAUSTED. REPRODUCTIONS
WILL BE MADE IN BLACK AND WHITE ONLY.

TSL-121-2/65

DATE PROCESSED: 22 June 65

PROCESSOR: E D. Jaeger

AFCRL-65-306

MAGNETIC TAPE COPIES OF MIT GEOPHYSICS PROGRAM SET II

(TIME SERIES PROGRAMS FOR THE IBM 709, 7090, 7094)

S. M. Simpson, Jr.

Massachusetts Institute of Technology
Cambridge 39, Massachusetts

Contract No. AF19(604)-7378

Project No. 8652

Task No. 865203

Scientific Report No. 10

March 31, 1965

Work Sponsored by Advanced Research Projects Agency

Project Vela-Uniform

ARPA Order No. 180-61, Amendment 2

Prepared
for

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS

Requests for additional copies by agencies of the Department of Defense, their contractors, or other government agencies should be directed to:

Defense Documentation Center (DDC)
Cameron Station
Alexandria, Virginia 22314

Department of Defense contractors must be established for DDC services or have their "need-to-know" certified by the cognizant military agency of their project or contract.

All other persons and organizations should apply to the:

Clearinghouse for Federal Scientific
and Technical Information (CFSTI)
Sills Building
5285 Port Royal Road
Springfield, Virginia 22151

ABSTRACT

The set of programs known as "MIT Geophysics Program Set I" has been expanded, edited, and upgraded to form Set II. This new set consists of 267 programs for the IBM 709, 7090, 7094 and is available to qualified applicants, via magnetic tape copies of the symbolic decks, from the Seismic Data Laboratory of United Electrodynamics. A complete copy requires two 2400 foot high density (900 BPI) tapes.

The symbolic decks of Set II form an interlocking system of self-documenting (including examples) subroutines written in FORTRAN and FAP (compatible with FORTRAN-II) concerned primarily with single and multiple time series analysis. Because of the subroutine nature of its construction, however, much of the system is readily accessible for use in other computational areas.

The new programs in Set II concentrate largely on utility functions (graphical and other input-output, miscellaneous numerical operators) and on time series operators for multidimensional and multi-input processes (including in particular high speed recursion techniques for solving least squares simultaneous equations). A handful of specialized or outmoded programs from Set I has been suppressed; most of the others have been upgraded with respect to documentation; and some have been modified with respect to coding.

TABLE OF CONTENTS

BLANK PAGE

1. Introduction

MIT Geophysics Program Set II is an expanded, modified version of Program Set I which was introduced (Simpson, 1962) as follows.

> "The MIT Department of Geology and Geophysics has a history in time series computations by high speed computers which extends back to 1952 when it began using Whirlwind I to instrument Wiener's optimum filter concepts in the signal-noise problems of reflection seismology. Since then it has steadily developed and expanded the computer technology of time series analysis, adapting computational concepts to the shifting ground of new machine languages.
>
> "The programs developed in this process have been made available on an individual basis in the past but, particularly with impetus from VELA UNIFORM research, the increased volume of requests have necessitated a more concentrated effort to systematize this distribution. Moreover, the widespread adoption of FORTRAN and IBM 700 series machines justifies for us the considerable effort we have taken to carefully document and assemble the large number of our most useful programs which we are now making available as "MIT Geophysics Program Set I.
>
> "Symbolic programs are the best for general distribution and because of the number of cards involved (over 23,000) we have chosen to transmit them by magnetic tape. The symbolic programs on the tape copies are completely self-explanatory. The present report is concerned with supplementary information such as complete tables of contents, conventions used in program design and description, details on the production and testing of the master tape, and a KWIC-type index to the programs.
>
> "The bulk of the programs included are the work of Stephen M. Simpson, Jr., Jon F. Claerbout, James N. Galbraith, and Ralph A. Wiggins, but they include contributions from Jacqueline Clark, Enders A. Robinson, Roy J. Greenfield, and there are a few programs originating in the MIT Computation Center as well as one or two modifications

of FORTRAN system routines. Authorship is given individually in the comment cards of each program.

"The production and testing of the master tape involved not only the work of the authors but also extensive test program writing by Joseph Procito and seemingly endless card preparation, handling and editing by Elizabeth Studer, Dauna Trop, and Karl Gentili to whom the authors are most grateful.

"Test computations were performed both on the IBM 7090 at the MIT Computation Center and on the IBM 709 of the Cooperative Computing Laboratory of MIT, with the valuable assistance of Michael Saxton and Anthony Sacco, respectively."

The above serves to introduce Program Set II with the following additional comments

1. The symbolic card count now exceeds 50,000.

2. The names of Mrs. Myrna Kasser, Regina Lahteine, and Mrs. Barbara Cullum should be added to the list of those assisting in punched card work and the names of John Harmon, Thomas Burhoe, Mason Fleming and William Jarvis to the list of computer operators.

3. The IBM 7094 of the MIT Computation Center was the principal computing instrument used during the period since Program Set I.

REFERENCE

Simpson, Jr., S. M., 1962, Magnetic tape copies of MIT Geophysics Program Set I (Time series programs for the IBM 709, 7090): Sci. Rept. 4 of Contract AF 19(604)7378, AFCRL-65-207, ARPA Project VELA UNIFORM.

2. Tables of Contents of the Symbolic Tapes

The symbolic versions of the 267 programs of Set II appear on two BCD tapes, 116 on the first tape and 151 on the second. The first file of each tape gives a table of contents for that tape, and the remaining files are the successive programs, ordered alphabetically by program name, terminated by an "END TAPE" file. Consequently the first tape contains 118 files and the second one 153 files. The following 11 pages show listings of the first files of the two tapes.

Listing of first file of Tape 1 of Program Set II (Page 1 of 5)

```
•                TABLE OF CONTENTS
•        FILE NO.   1 ON THIS TAPE IS
•TABLE OF CONTENTS
•        FILE NO.   2 ON THIS TAPE IS
•ABSVAL             FAST ABSOLUTE VALUE OF A VECTOR
•        FILE NO.   3 ON THIS TAPE IS
•ADANL              MODIFY AUTO- OR CROSS-CORRELATIONS FOR DANIELL SPECTRA
•        FILE NO.   4 ON THIS TAPE IS
•ADDK               MODIFY A SET OF VARIABLES BY A CONSTANT OR BY CONSTANTS
•        FILE NO.   5 ON THIS TAPE IS
•AMPHZ              AMPLITUDE AND PHASE FROM REAL AND IMAGINARY, OR REVERSE
•        FILE NO.   6 ON THIS TAPE IS
•ARBCCL             FIND A MATRIX COLUMN WITH ARBITRARY INDEX BY INTERPOLATION
•        FILE NO.   7 ON THIS TAPE IS
•ARCTAN             ARCTANGENT FUNCTION
•        FILE NO.   8 ON THIS TAPE IS
•ASPECT             FAST COSINE TRANSFORMS OF ONE-SIDED AUTOCORRELATIONS
•        FILE NO.   9 ON THIS TAPE IS
•ASPEC2             AUTOSPECTRUM BY COSINE TRANSFORM OF AUTOCORRELATION
•        FILE NO.  10 ON THIS TAPE IS
•AVRAGE             FIND AVERAGE OF FLOATING VECTOR
•        FILE NO.  11 ON THIS TAPE IS
•BLKSUM             SUMMATION OF VECTOR OVER ABUTTING BLOCKS OF CONSTANT LENGTH
•        FILE NO.  12 ON THIS TAPE IS
•BOOST              ADD A CONSTANT TO ELEMENTS OF A FXD OR FLTG VECTOR
•        FILE NO.  13 ON THIS TAPE IS
•CARIGE             SPACE CARRIAGE N LINES OR RESTORE PAGE
•        FILE NO.  14 ON THIS TAPE IS
•CHISQR             COMPUTE CHI-SQUARE FOR CONSTANT PROBABILITY CASE
•        FILE NO.  15 ON THIS TAPE IS
•CHOOSE             SET A LIST OF VARIABLES TO ONE OF TWO SETS OF VALUES
•        FILE NO.  16 ON THIS TAPE IS
•CHPRTS             FAST REVERSAL OF SPECIAL VECTORS (AS PRODUCED BY SPLIT)
•        FILE NO.  17 ON THIS TAPE IS
•CHSIGN             CHANGE ALL SIGN BITS OF A VECTOR
•        FILE NO.  18 ON THIS TAPE IS
•CLKON              CHECK IF INTERVAL TIMER IS ON MAKING ON-LINE REQUEST IF NOT
•        FILE NO.  19 ON THIS TAPE IS
•CLOCK1 (7090)      FOR REAL TIME TIMING IN SECONDS USING 7090 INTERVAL CLOCK
•        FILE NO.  20 ON THIS TAPE IS
•CMPARP             COMPARE PAIRS OF VARIABLES OR A SET OF VARIABLES FOR EQUALITY
•        FILE NO.  21 ON THIS TAPE IS
•CMPARV             FAST COMPARE TWO ARBITRARY MODE VECTORS FOR IDENTITY
•        FILE NO.  22 ON THIS TAPE IS
•CMPRA              COMPARE ARITHMETICALLY TWO WORDS WHERE -0 IS LESS THAN +0
•        FILE NO.  23 ON THIS TAPE IS
•CNTRC6             CONTOUR A MATRIX ON THE PRINTER IN DECIBELS
•        FILE NO.  24 ON THIS TAPE IS
•CNTRCW             FIND CONTOUR LEVELS FOR PLOTTING A ROW OF DATA
•        FILE NO.  25 ON THIS TAPE IS
•COLABL             LABEL PRINTER COLUMNS WITH INCREASING 3-DIGIT INTEGERS
•        FILE NO.  26 ON THIS TAPE IS
•COLAPS             COLLAPSE ONE-SIDED VECTOR INTO SMALLER RANGE
•        FILE NO.  27 ON THIS TAPE IS
•CONTLR             CONTOUR OF MATRIX SUBSET ON OFF-LINE PRINTER
```

Listing of first file of Tape 1 of
Program Set II (Page 2 of 5)

```
*       FILE NO.  28 ON THIS TAPE IS
*CONVLV           COMPLETE CONVOLUTION OF TWO TRANSIENTS
*       FILE NO.  29 ON THIS TAPE IS
*CONVLV-II        COMPLETE CONVOLUTION OF TWO TRANSIENTS
*       FILE NO.  30 ON THIS TAPE IS
*COSIS            FAST COSINE AND/OR SINE TRANSFORMS OF ODD-LENGTH SERIES
*       FILE NO.  31 ON THIS TAPE IS
*COSP             FAST COSINE AND/OR SINE TRANSFORMS FROM 2 OR 4 EVEN-ODD PARTS
*       FILE NO.  32 ON THIS TAPE IS
*COSTBL           GENERATE COSINE OR SINE HALF-WAVE TABLES, FIXED OR FLOATING
*       FILE NO.  33 ON THIS TAPE IS
*CPYFL2           FAST COPY FILE FROM ONE TAPE TO ANOTHER - VERSION 2
*       FILE NO.  34 ON THIS TAPE IS
*CROSS            CROSSCORRELATION OF TRANSIENTS BEGINNING WITH ZERO LAG
*       FILE NO.  35 ON THIS TAPE IS
*CROST            CROSSCORRELATION OF TRANSIENTS BEGINNING WITH ANY LAG
*       FILE NO.  36 ON THIS TAPE IS
*CRSVM            CROSSCORRELATION OF TRANSIENT VECTORS OF MATRICES
*       FILE NO.  37 ON THIS TAPE IS
*CSOUT            OUTPUT VARIABLES FIVE PER LINE IN G FORMAT
*       FILE NO.  38 ON THIS TAPE IS
*CUFIT1           FIND CUBIC WHICH EXACTLY FITS 4 EQUALLY     ED POINTS
*       FILE NO.  39 ON THIS TAPE IS
*CVSOUT           OUTPUT COLUMN VECTORS BY NORMAL OR LITERAL FORMATS
*       FILE NO.  40 ON THIS TAPE IS
*DADECK           LIST DATA DECK AND REPOSITION TAPE TO FRONT OF DECK
*       FILE NO.  41 ON THIS TAPE IS
*DELTA            DELTA FUNCTION AND STEP FUNCTIONS, FLOATING AND FIXED POINT
*       FILE NO.  42 ON THIS TAPE IS
*DERIVA           DERIVATIVE OF A VECTOR BY DIFFERENCING
*       FILE NO.  43 ON THIS TAPE IS
*DIFPRS           DIFFERENCE FIXED OR FLOATING VECTOR ELEMENTS IN PAIRS
*       FILE NO.  44 ON THIS TAPE IS
*DISPLA (709)     WRITE HOLLERITH TEXT ON SCOPE
*       FILE NO.  45 ON THIS TAPE IS
*DISPLA(7090)     WRITE HOLLERITH TEXT ON SCOPE
*       FILE NO.  46 ON THIS TAPE IS
*DIVIDE           DIVIDE A FLOATING VECTOR BY A CONSTANT
*       FILE NO.  47 ON THIS TAPE IS
*DOTJ             VECTOR DOT PRODUCT WITH ARBITRARY INCREMENTS
*       FILE NO.  48 ON THIS TAPE IS
*DOTP             DISPLACED DOT PRODUCT OF 2-DIMENSIONAL ARRAYS
*       FILE NO.  49 ON THIS TAPE IS
*DSPFMT           VARIABLE ORIGIN FORMAT GENERATOR FOR SCOPE SUBROUTINE DISPLA
*       FILE NO.  50 ON THIS TAPE IS
*DUBLX            FAST DOUBLING OR HALVING OF A VECTOR (FIXED OR FLOATING)
*       FILE NO.  51 ON THIS TAPE IS
*EXCHVS           EXCHANGE ANY TWO VECTORS
*       FILE NO.  52 ON THIS TAPE IS
*EXPAND           HI-SPEED EXPANSION OF A VECTOR UNDER CUBIC INTERPOLATION
*       FILE NO.  53 ON THIS TAPE IS
*FACTOR           FACTOR POWER SPECTRUM TO FIND MINIMUM PHASE WAVELET
*       FILE NO.  54 ON THIS TAPE IS
*FAPSUM           COMPUTE A LOGICAL SUMCHECK
```

Listing of first file of Tape 1 of
Program Set II (Page 3 of 5)

```
*       FILE NO.  55 ON THIS TAPE IS
*FASCN1           FAST SCAN VECTOR FOR ELEMENT EQUAL OR GREATER THAN GIVEN VALUE
*       FILE NO.  56 ON THIS TAPE IS
*FASCUB           FAST EVALUATE CUBIC FOR EVENLY SPACED ARGUMENTS
*       FILE NO.  57 ON THIS TAPE IS
*FASTRK           FAST TRACK THROUGH A VECTOR OF INDICES
*       FILE NO.  58 ON THIS TAPE IS
*FDOT             FAST DOT PRODUCT OF TWO VECTORS
*       FILE NO.  59 ON THIS TAPE IS
*FIRE2            TWO-DIMENSIONAL FILTER BY RECURSION
*       FILE NO.  60 ON THIS TAPE IS
*FIXV             FIX A FLOATING VECTOR WITH OR WITHOUT ROUNDING
*       FILE NO.  61 ON THIS TAPE IS
*FLOATM           FLOAT ANY MACHINE LANGUAGE INTEGER
*       FILE NO.  62 ON THIS TAPE IS
*FLOATV           FLOAT A VECTOR
*       FILE NO.  63 ON THIS TAPE IS
*FMTOUT           WRITE OUTPUT TAPE WITH NORMAL OR LITERAL FORMAT VECTOR
*       FILE NO.  64 ON THIS TAPE IS
*FNDFMT           ACCESS TO LITERAL OR ORDINARY FORMAT
*       FILE NO.  65 ON THIS TAPE IS
*FRAME  (709)     ADVANCE FILM FRAME ON SCOPE
*       FILE NO.  66 ON THIS TAPE IS
*FRAME(7090)      ADVANCE FILM FRAME ON SCOPE
*       FILE NO.  67 ON THIS TAPE IS
*FRQCT1           FREQUENCY DISTRIBUTION OF A FIXED POINT VECTOR
*       FILE NO.  68 ON THIS TAPE IS
*FRQCT2           FREQUENCY COUNT OF NUMBER OF VALUES OF A SERIES IN GIVEN RANGES
*       FILE NO.  69 ON THIS TAPE IS
*FSKIP            SKIP FORWARD OR BACKWARD OVER FILES ON TAPE
*       FILE NO.  70 ON THIS TAPE IS
*FT24             HIGH SPEED 24 POINT SPECTRUM
*       FILE NO.  71 ON THIS TAPE IS
*FT24  -II        HIGH SPEED 24 POINT SPECTRUM
*       FILE NO.  72 ON THIS TAPE IS
*FXDATA           SCALE, CONVERT FLTG. VECTOR TO MACHINE INTEGERS OR CONVERSELY
*       FILE NO.  73 ON THIS TAPE IS
*GENHOL           GENERATE HOLLERITH FIELD
*       FILE NO.  74 ON THIS TAPE IS
*GETHOL           GET HOLLERITH DATA FROM CALLING SEQUENCE
*       FILE NO.  75 ON THIS TAPE IS
*GETRD1           ACCESS ROUTINE FOR RAND CORP. MILLION RANDOM DIGITS FROM TAPE
*       FILE NO.  76 ON THIS TAPE IS
*GETX             ALLOWS VARIABLE DEPTH INDEXING OF VECTORS
*       FILE NO.  77 ON THIS TAPE IS
*GNFLT1           GENERATE SYMMETRICAL FILTER WITH GIVEN AMPLITUDE RESPONSE
*       FILE NO.  78 ON THIS TAPE IS
*GNHOL2           GENERATE HOLLERITH CHARACTERS
*       FILE NO.  79 ON THIS TAPE IS
*GRAPH            MULTIPLE FRAME SCOPE PLOTS OF VECTOR SETS
*       FILE NO.  80 ON THIS TAPE IS
*GRAPHX           SUBROUTINE GRAPH EXPANDED OVER VERTICAL FRAMES
*       FILE NO.  81 ON THIS TAPE IS
*GRUP2            DIVIDE THE X AXIS INTO EQUALLY PROBABLE RANGES
```

Listing of first file of Tape 1 of
Program Set II (Page 4 of 5)

```
*       FILE NO.  82 ON THIS TAPE IS
*HLADJ            HOLLERITH LEFT ADJUST OR RIGHT ADJUST FUNCTION
*       FILE NO.  83 ON THIS TAPE IS
*HSTPLT           HISTOGRAM PLOTTING FOR SUBROUTINE GRAPH
*       FILE NO.  84 ON THIS TAPE IS
*HSTPLT-II        BAR GRAPH PLOTTING FOR SUBROUTINE GRAPH
*       FILE NO.  85 ON THIS TAPE IS
*HSTPLT-III(709) CUBIC CURVE SCOPE PLOTTING FOR SUBROUTINE GRAPH
*       FILE NO.  86 ON THIS TAPE IS
*HSTPLT-III(7090)CUBIC CURVE SCOPE PLOTTING FOR SUBROUTINE GRAPH
*       FILE NO.  87 ON THIS TAPE IS
*HVTOIV           SPREAD OUT HOLLERITH VECTOR AS FORTRAN INTEGERS
*       FILE NO.  88 ON THIS TAPE IS
*IDERIV           INVERSION OF DIFFERENTIATION BY DIFFERENCING
*       FILE NO.  89 ON THIS TAPE IS
*IFNCTN           INVERSION OF A MONOTONE FUNCTION BY LINEAR INTERPOLATION
*       FILE NO.  90 ON THIS TAPE IS
*IINTGR           INVERSION OF TRAPEZOIDAL INTEGRAL
*       FILE NO.  91 ON THIS TAPE IS
*INDATA           FAST AND CONVENIENT RETRIEVAL OF DATA FROM A SPECIAL TAPE
*       FILE NO.  92 ON THIS TAPE IS
*INDEX            HYBRID SUBPROGRAMS FOR INCREMENTING, TESTING, AND SETTING
*       FILE NO.  93 ON THIS TAPE IS
*INTGRA           INDEFINITE INTEGRAL BY TRAPEZOIDAL RULE
*       FILE NO.  94 ON THIS TAPE IS
*INTHCL           INTERPRET HOLLERITH
*       FILE NO.  95 ON THIS TAPE IS
*INTOPR           INTERPOLATION OPERATOR FOR 1 TO 4 EVENLY SPACED DATA VALUES
*       FILE NO.  96 ON THIS TAPE IS
*INTSUM           INTEGRATED SUMMATION OF A FLOATING OF FIXED VECTOR
*       FILE NO.  97 ON THIS TAPE IS
*IPLYEV           COMPLEX POLYNOMIAL EVALUATION
*       FILE NO.  98 ON THIS TAPE IS
*ITOMLI           FAST CONVERT FORTRAN INTEGER VECTOR TO MLI VECTOR
*       FILE NO.  99 ON THIS TAPE IS
*IVTOHV           PACK UP FORTRAN INTEGER VECTOR AS HOLLERITH VECTOR
*       FILE NO. 100 ON THIS TAPE IS
*IXCARG           LOCATE ARGUMENT WITH RESPECT TO COMMON
*       FILE NO. 101 ON THIS TAPE IS
*KIINTI           PROBABILITY THAT A CHI-SQUARED VARIATE EXCEEDS A VALUE
*       FILE NO. 102 ON THIS TAPE IS
*KOLAPS           COLLAPSE ODD-LENGTHED VECTOR ABOUT ITS MIDPOINT
*       FILE NO. 103 ON THIS TAPE IS
*LIMITS           CHECK THAT VARIABLES FROM LIST FALL WITHIN GIVEN LIMITS
*       FILE NO. 104 ON THIS TAPE IS
*LINE (709)       FAST ARBITRARY STRAIGHT LINE SEGMENT ON SCOPE
*       FILE NO. 105 ON THIS TAPE IS
*LINE (7090)      FAST ARBITRARY STRAIGHT LINE SEGMENT ON SCOPE
*       FILE NO. 106 ON THIS TAPE IS
*LINEH  (709)     PLOT FAST HORIZONTAL LINE ON SCOPE
*       FILE NO. 107 ON THIS TAPE IS
*LINEH(7090)      PLOT FAST HORIZONTAL LINE ON SCOPE
*       FILE NO. 108 ON THIS TAPE IS
*LINEV  (709)     PLOT FAST VERTICAL LINE ON SCOPE
```

Listing of first file of Tape 1 of
Program Set II (Page 5 of 5)

```
*       FILE NO. 109 ON THIS TAPE IS
*LINEV(7090)     PLOT FAST VERTICAL LINE ON SCOPE
*       FILE NO. 110 ON THIS TAPE IS
*LINTR1          LINEAR INTERPOLATION IN A TABLE
*       FILE NO. 111 ON THIS TAPE IS
*LISTNG          LIST AUXILIARY INFORMATION FOR AN INDATA-OUDATA TYPE TAPE
*       FILE NO. 112 ON THIS TAPE IS
*LOC             CORE LOCATION WITH INDEXABLE ARGUMENT
*       FILE NO. 113 ON THIS TAPE IS
*LOCATE          LOCATE AND OPERATE SUBROUTINES BY PROXY CALL STATEMENTS
*       FILE NO. 114 ON THIS TAPE IS
*LSHFT           LOGICAL SHIFT FUNCTION
*       FILE NO. 115 ON THIS TAPE IS
*LSLINE          LEAST SQUARES LINE
*       FILE NO. 116 ON THIS TAPE IS
*LSSS1           LEAST SQUARES SHAPER BY SIDEWAYS ITERATION
*       FILE NO. 117 ON THIS TAPE IS
*MATINV          INVERSE OF A MATRIX
*       FILE NO. 118 ON THIS TAPE IS
*END TAPE CARD IN FORMAT(1H*,6X,8HEND TAPE)
```

Listing of first file of Tape 2 of
Program Set II (Page 1 of 6)

```
*                TABLE OF CONTENTS
*       FILE NO.   1 ON THIS TAPE IS
*TABLE OF CONTENTS
*       FILE NO.   2 ON THIS TAPE IS
*MATML1           SQUARE MATRIX MULTIPLICATION
*       FILE NO.   3 ON THIS TAPE IS
*MATML3           N X M MATRIX BY M X L MATRIX MULTIPLICATION
*       FILE NO.   4 ON THIS TAPE IS
*MATRA            MATRIX TRANSPOSE
*       FILE NO.   5 ON THIS TAPE IS
*MATRA1           SQUARE MATRIX TRANSPOSE
*       FILE NO.   6 ON THIS TAPE IS
*MAXSN            FIND SIGNED OR UNSIGNED EXTREMAL VALUES OF A VECTOR
*       FILE NO.   7 ON THIS TAPE IS
*MAXSNM           EXTREMAL VALUES OF MATRIX ELEMENTS
*       FILE NO.   8 ON THIS TAPE IS
*MDOT             DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATRICES
*       FILE NO.   9 ON THIS TAPE IS
*MDOT3            DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATRICES
*       FILE NO.  10 ON THIS TAPE IS
*MEMUSE           OFF-LINE PRINT OF MEMORY USAGE - PROGRAM AND COMMON
*       FILE NO.  11 ON THIS TAPE IS
*MFACT            FACTOR A SYMMETRIC POSITIVE DEFINITE MATRIX
*       FILE NO.  12 ON THIS TAPE IS
*MIFLS            MULTI-INPUT FILTER BY LEAST SQUARES
*       FILE NO.  13 ON THIS TAPE IS
*MIPLS            MULTI-INPUT PREDICTOR BY LEAST SQUARES
*       FILE NO.  14 ON THIS TAPE IS
*MISS             MULTI-INPUT SIDEWARDS ITERATION
*       FILE NO.  15 ON THIS TAPE IS
*MLISCL           MULTIPLY AN MLI VECTOR BY A FORTRAN FIXED POINT INTEGER
*       FILE NO.  16 ON THIS TAPE IS
*MLI2A6           CONVERT MACHINE LANGUAGE INTEGER TO EQUIVALENT HOLLERITH
*       FILE NO.  17 ON THIS TAPE IS
*MONOCK           CHECK VECTOR FOR MONOTONE INCREASING OR DECREASING BEHAVIOR
*       FILE NO.  18 ON THIS TAPE IS
*MOUT             MATRIX OUTPUT IN G FORMAT
*       FILE NO.  19 ON THIS TAPE IS
*MOUTAI           OUTPUT A MATRIX AS INTEGERS DENSELY PACKED OFF-LINE
*       FILE NO.  20 ON THIS TAPE IS
*MOVE             MOVE A VECTOR TO A DIFFERENT LOCATION
*       FILE NO.  21 ON THIS TAPE IS
*MOVECS           MOVE AN ARBITRARY SET OF VECTORS
*       FILE NO.  22 ON THIS TAPE IS
*MOVREV           MOVE,REVERSE, CHANGE SPACING, OR CHANGE SIGN OF A VECTOR
*       FILE NO.  23 ON THIS TAPE IS
*MPSEQ1           MAP A SEQUENCE OF NUMBERS INTO AN INTEGER SERIES
*       FILE NO.  24 ON THIS TAPE IS
*MRVRS            REVERSE VECTOR OF MATRICES
*       FILE NO   25 ON THIS TAPE IS
*MSCON1           MEAN SQUARE CONTINGENCY AND DEPENDENCY FROM PROBABILITY DENSITY
*       FILE NO.  26 ON THIS TAPE IS
*MULK  -II        MULTIPLY ANY NO. OF VARIABLES BY A SINGLE FLTG. PT. CONSTANT
*       FILE NO.  27 ON THIS TAPE IS
*MULLER           POLYNOMIAL ROOT FINDER
```

Listing of first file of Tape 2 of
Program Set II (Page 2 of 6)

```
*       FILE NO.  28 ON THIS TAPE IS
*MULPLY           MULTIPLY VECTOR BY FLOATING OR FIXED CONSTANT
*       FILE NO.  29 ON THIS TAPE IS
*MUVACD           FAST MOVING SUMMATION OF A FIXED POINT VECTOR
*       FILE NO.  30 ON THIS TAPE IS
*MVBLCK           MOVE DATA BLOCK
*       FILE NO.  31 ON THIS TAPE IS
*MVINAV           MOVING AVERAGE OF A VECTOR
*       FILE NO.  32 ON THIS TAPE IS
*MVNSUM           MOVING SUMMATION WITH DIVISION BY A CONSTANT
*       FILE NO.  33 ON THIS TAPE IS
*MVNTIN           MOVING TRAPEZOIDAL INTEGRAL OR ABSOLUTE VALUE INTEGRAL
*       FILE NO.  34 ON THIS TAPE IS
*MVSQAV           MOVING MEAN SQUARE AVERAGE OF A VECTOR
*       FILE NO.  35 ON THIS TAPE IS
*MXRARE           REGION TO MAXIMIZE RATIO OF TWO DISTRUBUTION FUNCTIONS
*       FILE NO.  36 ON THIS TAPE IS
*NMZMG1           NORMALIZE A VECTOR TO GIVEN MAXIMUM VALUE
*       FILE NO.  37 ON THIS TAPE IS
*NOINT1           NORMAL DISTRIBUTION AND DIVISION INTO EQUALLY LIKELY SECTIONS
*       FILE NO.  38 ON THIS TAPE IS
*NRMVEC           NORMALIZE AND CHANGE MEAN OF A VECTOR
*       FILE NO.  39 ON THIS TAPE IS
*NTHA             RETURN N-TH ARGUMENT BEYOND THE FIRST
*       FILE NO.  40 ON THIS TAPE IS
*NURINC           CREATE ONE VECTOR FROM ANOTHER WITH NEW RANGE AND INCREMENT
*       FILE NO.  41 ON THIS TAPE IS
*NXALRM           SCAN VECTOR FOR POSSIBLE BLOCK OF VALUES ALL ABOVE GIVEN LEVEL
*       FILE NO.  42 ON THIS TAPE IS
*ONLINE           OPTIONAL ONLINE MONITOR OF BCD TAPE WRITING
*       FILE NO.  43 ON THIS TAPE IS
*OUDATA           FAST AND CONVENIENT DATA STORAGE ON TAPE
*       FILE NO.  44 ON THIS TAPE IS
*PACDAT           READ EVERY N-TH WORD FROM BINARY TAPE
*       FILE NO.  45 ON THIS TAPE IS
*PAKN             SCALE AND FIX DATA VECTOR, PACK N DATA POINTS PER REGISTER
*       FILE NO.  46 ON THIS TAPE IS
*PLANSP           FAST TWO-DIMENSIONAL SPATIAL SPECTRUM
*       FILE NO.  47 ON THIS TAPE IS
*PLOTVS           PRINTER-PLOT OF ARBITRARY SET OF VECTORS
*       FILE NO.  48 ON THIS TAPE IS
*PLTVS1           PRINTER PLOT OF A SET OF EQUAL LENGTH VECTORS
*       FILE NO.  49 ON THIS TAPE IS
*PLURAS           PLURALIZE THE NEXT SUBROUTINE
*       FILE NO.  50 ON THIS TAPE IS
*PLYSYN           POLYNOMIAL SYNTHESIZED FROM ITS REAL AND COMPLEX ROOTS
*       FILE NO.  51 ON THIS TAPE IS
*POKCT1           EVALUATION OF INTEGER SEQUENCE IN GROUPS OF FIVE AS POKER HANDS
*       FILE NO.  52 ON THIS TAPE IS
*POLYDV           PERFORM LONG DIVISION OF TWO POLYNOMIALS
*       FILE NO.  53 ON THIS TAPE IS
*POLYEV           EVALUATE A POLYNOMIAL WITH REAL COEFFICIENTS FOR REAL ARGUMENT
*       FILE NO.  54 ON THIS TAPE IS
*POLYSN           POLYNOMIAL SYNTHESIS FROM REAL AND COMPLEX ROOTS
```

Listing of first file of Tape 2 of
Program Set II (Page 3 of 6)

```
*       FILE NO.  55 ON THIS TAPE IS
*POWER          RAISE VECTOR TO POWER OR SUM POWER OF DEVIATIONS FROM BASE
*       FILE NO.  56 ON THIS TAPE IS
*PRBFIT         GENERATE PROBABILITY DISTRIBUTION WITH SPECIFIED MOMENTS
*       FILE NO.  57 ON THIS TAPE IS
*PROB2          SECOND PROBABILITY DENSITY OF INTEGER SERIES AT GIVEN LAG
*       FILE NO.  58 ON THIS TAPE IS
*PROCCR         FAST CORRELATIONS FOR LONG SERIES OF FIXED POINT INTEGERS
*       FILE NO.  59 ON THIS TAPE IS
*PSQRT          FIND THE POWER SERIES SQUARE ROOT OF A POLYNOMIAL
*       FILE NO.  60 ON THIS TAPE IS
*PWMLIV         PRINT OR WRITE OUTPUT TAPE A MACHINE LANGUAGE INTEGER VECTOR
*       FILE NO.  61 ON THIS TAPE IS
*QACORR         FAST AUTOCORRELATIONS FOR LONG, LIMITED ACCURACY SERIES
*       FILE NO.  62 ON THIS TAPE IS
*QCNVLV         FAST CONVOLUTIONS FOR LONG, LIMITED ACCURACY SERIES
*       FILE NO.  63 ON THIS TAPE IS
*QFURRY         FAST FOURIER TRANSFORM OF TRANSIENT WITH ARBITRARY TIME ORIGIN
*       FILE NO.  64 ON THIS TAPE IS
*QIFURY         QUICK INVERSE FOURIER TRANSFORM WITH ARBITRARY TIME ORIGIN
*       FILE NO.  65 ON THIS TAPE IS
*QINTRI         QUADRATIC INTERPOLATION IN A TABLE
*       FILE NO.  66 ON THIS TAPE IS
*QUFITI         FIND QUADRATIC WHICH EXACTLY FITS 3 EQUALLY SPACED POINTS
*       FILE NO.  67 ON THIS TAPE IS
*QXCORR         FAST CROSS-CORRELATIONS FOR LONG, LIMITED ACCURACY SERIES
*       FILE NO.  68 ON THIS TAPE IS
*QXCORI         QUICK CROSSCORRELATION OF MLI TRANSIENTS
*       FILE NO.  69 ON THIS TAPE IS
*RDATA          READ DATA IN GENERALIZED FORMAT
*       FILE NO.  70 ON THIS TAPE IS
*REFLEC         REFLECT A FIXED OR FLOATING VECTOR THROUGH A CONSTANT
*       FILE NO.  71 ON THIS TAPE IS
*REMAV          REMOVE THE MEAN FROM A FLOATING VECTOR
*       FILE NO.  72 ON THIS TAPE IS
*REREAD         REREAD DATA RECORD AND END FILE MONITOR
*       FILE NO.  73 ON THIS TAPE IS
*REVER          REVERSE A VECTOR ELSEWHERE OR IN PLACE
*       FILE NO.  74 ON THIS TAPE IS
*REVERS         FAST REVERSE STORAGE ORDER OF A VECTOR
*       FILE NO.  75 ON THIS TAPE IS
*RLSPR          REALIZABLE LEAST SQUARES PREDICTOR BY RECURSION, 1-DIMENSION
*       FILE NO.  76 ON THIS TAPE IS
*RLSPR2         REALIZABLE LEAST SQUARES PREDICTOR BY RECURSION, 2-DIMENSIONS
*       FILE NO.  77 ON THIS TAPE IS
*RLSSR          REALIZABLE LEAST SQUARES SHAPER BY RECURSION
*       FILE NO.  78 ON THIS TAPE IS
*RMSDEV         R.M.S. DEVIATION FROM GIVEN BASE OR FROM TRUE AVERAGE
*       FILE NO.  79 ON THIS TAPE IS
*RND            ROUND FLTG. PT. NO. UP, DOWN, OR TO NEAREST FLTG. PT. INTEGER
*       FILE NO.  80 ON THIS TAPE IS
*RNDV           ROUND, ROUND UP, OR ROUND DOWN A FLOATING VECTOR
*       FILE NO.  81 ON THIS TAPE IS
*ROAR2          ROTATE CENTRO-SYMMETRIC O-   TISYMMETRIC 2-DIMENSIONAL ARRAY
```

```
*       FILE NO.  82 ON THIS TAPE IS
*ROTATI           ROTATE A VECTOR UPWARDS OR DOWNWARDS AN ARBITRARY AMOUNT
*       FILE NO.  83 ON THIS TAPE IS
*RPLFPT           REPLACE THE FORMAT OF A SUCCEEDING INPUT OR OUTPUT STATEMENT
*       FILE NO.  84 ON THIS TAPE IS
*RSKIP            SKIP FORWARD OR BACKWARD OVER RECORDS ON TAPE
*       FILE NO.  85 ON THIS TAPE IS
*SAME             ENABLE MIXED EXPRESSIONS IN FORTRAN
*       FILE NO.  86 ON THIS TAPE IS
*SCPSCL           SCALE VECTOR TO INTEGERS FOR SCOPE, CLIPPING EXCESSIVE VALUES
*       FILE NO.  87 ON THIS TAPE IS
*SEARCH           SEARCH A VECTOR FOR A VALUE
*       FILE NO.  88 ON THIS TAPE IS
*SEQSAC           FAST FUNCTIONS FOR SEQUENTIAL SINES AND COSINES
*       FILE NO.  89 ON THIS TAPE IS
*SETINO           INITIALIZE FOR ADDING TO AN INDATA-OUDATA TAPE
*       FILE NO.  90 ON THIS TAPE IS
*SETK             SET VARIABLES OR VECTORS TO GIVEN VALUES
*       FILE NO.  91 ON THIS TAPE IS
*SETK  -II        SET ANY NO. OF VARIABLES EQUAL TO A SINGLE VALUE (FXD OR FLTG)
*       FILE NO.  92 ON THIS TAPE IS
*SETKP            PLURALIZED FORMS OF SUBROUTINES SETK AND SETVEC
*       FILE NO.  93 ON THIS TAPE IS
*SETKS -II        SET ANY NO. OF VARIABLES EQUAL TO SEPARATE VALUES (FXD OR FLTG)
*       FILE NO.  94 ON THIS TAPE IS
*SETKV            SET ALL ELEMENTS OF VECTOR EQUAL TO A CONSTANT (ANY MODE)
*       FILE NO.  95 ON THIS TAPE IS
*SETKVS           SET ANY NO. OF VECTORS EQUAL TO SEPARATE VALUES (FXD OR FLTG)
*       FILE NO.  96 ON THIS TAPE IS
*SETLIN           SET FXD OR FLTG VECTOR EQUAL TO A LINEAR SEGMENT
*       FILE NO.  97 ON THIS TAPE IS
*SETLNS           SET LINEAR VECTORS, FIXED AND/OR FLOATING
*       FILE NO.  98 ON THIS TAPE IS
*SEVRAL           OPERATE SEVERAL SUBROUTINES OR ONE SUBROUTINE REPEATEDLY
*       FILE NO.  99 ON THIS TAPE IS
*SHFTR1           SHIFT VECTOR ELEMENTS ARITHMETICALLY LEFT OR RIGHT
*       FILE NO. 100 ON THIS TAPE IS
*SHFTR2           SHIFT VECTOR ELEMENTS LOGICALLY LEFT OR RIGHT
*       FILE NO. 101 ON THIS TAPE IS
*SHUFFL           SHUFFLE A LIST OF INTEGERS FROM 1 TO N
*       FILE NO. 102 ON THIS TAPE IS
*SIFT             FORM A VECTOR BY SIFTING ANOTHER AT EVEN INCREMENTS
*       FILE NO. 103 ON THIS TAPE IS
*SIMEQ            SOLUTION OF SIMULTANEOUS EQUATIONS AND DETERMINANT EVALUATION
*       FILE NO. 104 ON THIS TAPE IS
*SIZELP           FAST MAKE INDEX (BY INCREASING SIZE) OF ELEMENTS IN A VECTOR
*       FILE NO. 105 ON THIS TAPE IS
*SMPSCN           UNSCALE OR SCALE VECTOR FOR SIMPSON INTEGRAL AND/OR INTEGRATE
*       FILE NO. 106 ON THIS TAPE IS
*SPCOR2           SPATIAL CROSSCORRELATION OF 2-DIMENSIONAL SPATIAL ARRAYS
*       FILE NO. 107 ON THIS TAPE IS
*SPLIT            SPLIT A VECTOR INTO ITS EVEN AND ODD PARTS (OR INVERSE)
*       FILE NO. 108 ON THIS TAPE IS
*SQRDFR           SUM SQUARE DIF. OF FLTG VECTOR FROM ANOTHER OR FROM A CONSTANT
```

Listing of first file of Tape 2 of

Program Set II (Page 5 of 6)

```
*       FILE NO. 109 ON THIS TAPE IS
*SQRMLI          FAST SQUARE ELEMENTS OF A MACHINE LANGUAGE INTEGER VECTOR
*       FILE NO. 110 ON THIS TAPE IS
*SQROOT          SQUARE ROOT OF A FLOATING VECTOR
*       FILE NO. 111 ON THIS TAPE  S
*SQRSUM          SUM THE SQUARED ELEMENTS OF A FLTG OR FXD VECTOR
*       FILE NO. 112 ON THIS TAPE IS
*SQUARE          SQUARE ELEMENTS OF FXD OR FLTG VECTOR
*       FILE NO. 113 ON THIS TAPE IS
*SRCH1           SEARCH VECTOR FOR NUMBER, STARTING FROM FIRST OR LAST TERM
*       FILE NO. 114 ON THIS TAPE IS
*STZ             FAST SET VECTOR TO ZERO
*       FILE NO. 115 ON THIS TAPE IS
*STZS            SET A LIST OF VECTORS TO ZERO
*       FILE NO. 116 ON THIS TAPE IS
*SUM             SUM ELEMENTS OF FLOATING OR FIXED VECTOR
*       FILE NO. 117 ON THIS TAPE IS
*SUMDFR          SUM DIFFERENCE OF VECTOR FROM ANOTHER OR FROM A CONSTANT
*       FILE NO. 118 ON THIS TAPE IS
*SWITCH          TEST THE CONDITION OF ANY SENSE SWITCH
*       FILE NO. 119 ON THIS TAPE IS
*TAMVL           TRIANGULAR AVERAGING, MOVING LEFT OR RIGHT END
*       FILE NO. 120 ON THIS TAPE IS
*TIMA2B (7094)   REAL TIME, TO SPECIFIED ACCURACY, OF GIVEN PROGRAM RANGE
*       FILE NO. 121 ON THIS TAPE IS
*TIMSUB          FIND OPERATION TIME OF NEXT SUBROUTINE TO GIVEN ACCURACY
*       FILE NO. 122 ON THIS TAPE IS
*TINGL           DEFINITE TRAPEZOIDAL INTEGRAL OF FUNCTION OR ITS MAGNITUDE
*       FILE NO. 123 ON THIS TAPE IS
*TRMINO          TERMINATE AN INDATA-OUDATA TAPE
*       FILE NO. 124 ON THIS TAPE IS
*UNPAKN          UNPACK AND RESCALE A PACKED DATA VECTOR
*       FILE NO. 125 ON THIS TAPE IS
*VARARG          ENABLE FORTRAN VARIABLE LENGTH CALLING SEQUENCES
*       FILE NO. 126 ON THIS TAPE IS
*VDOTV           DOT PRODUCT OF TWO VECTORS WITH DIVISION BY CONSTANT
*       FILE NO. 127 ON THIS TAPE IS
*VDVBYV          DIVIDE ELEMENTS OF ONE VECTOR BY THOSE OF ANOTHER
*       FILE NO. 128 ON THIS TAPE IS
*VECOUT          OFFLINE VECTOR OUTPUT WITH NORMAL OR LITERAL FORMAT
*       FILE NO. 129 ON THIS TAPE IS
*VOUT            OUTPUT NAMED VECTOR BY NORMAL OR LITERAL FORMAT WITH SPACING
*       FILE NO. 130 ON THIS TAPE IS
*VPLUSV          ADD OR SUBTRACT TWO FLOATING OR FIXED VECTORS
*       FILE NO. 131 ON THIS TAPE IS
*VRSOUT          OUTPUT VARIABLES BY NORMAL OR LITERAL FORMAT
*       FILE NO. 132 ON THIS TAPE IS
*VSOUT           OUTPUT NAMED VECTORS BY NORMAL OR LITERAL FORMATS WITH SPACING
*       FILE NO. 133 ON THIS TAPE IS
*VTIMSV          MULTIPLY ELEMENTS OF TWO VECTORS FIXED OR FLOATING
*       FILE NO. 134 ON THIS TAPE IS
*WAC             WIENER AUTOCORRELATION
*       FILE NO. 135 ON THIS TAPE IS
*WHICH           CHOOSE BETWEEN TWO VARIABLES BY A THIRD ONE BEING ZERO
```

```
*       FILE NO. 136 ON THIS TAPE IS
*WLLSFP          WIENER-LEVINSON LEAST SQUARE ERROR FILTER OR PREDICTOR
*       FILE NO. 137 ON THIS TAPE IS
*WRTOAT          WRITE BINARY DATA ON TAPE
*       FILE NO. 138 ON THIS TAPE IS
*XACTEQ          SIGN OF DIFFERENCE OF 2 VARIABLES OR 0 IF SAME INCLUDING SIG
*       FILE NO. 139 ON THIS TAPE IS
*XAVRGE          FIND AVERAGE OF FIXED PT VECTOR
*       FILE NO. 140 ON THIS TAPE IS
*XDIV            FXD PT DIVIDE WITH TRUNCATION OR ROUNDING TO FORTRAN-II INTEGER
*       FILE NO. 141 ON THIS TAPE IS
*XDVICE          DIVIDE A FXD VECTOR BY A CONSTANT
*       FILE NO. 142 ON THIS TAPE IS
*XFIXM           TRUNCATE OR ROUND FLOATING PT. NUMBER TO MACHINE INTEGER
*       FILE NO. 143 ON THIS TAPE IS
*XLCOMN          FIND LENGTH OF COMMON STORAGE
*       FILE NO. 144 ON THIS TAPE IS
*XLIMIT          FIND IF ARGUMENT FALLS INSIDE TWO LIMITING VALUES
*       FILE NO. 145 ON THIS TAPE IS
*XLOCV           CREATE VECTOR OF MACHINE ADDRESSES OF VARIABLES IN A LIST
*       FILE NO. 146 ON THIS TAPE IS
*XODZE           DETERMINE WHETHER FORTRAN-II INTEGER IS EVEN OR ODD
*       FILE NO. 147 ON THIS TAPE IS
*XREMAV          REMOVE THE MEAN FROM A FIXED VECTOR
*       FILE NO. 148 ON THIS TAPE IS
*XSPECT          FAST COSINE, SINE TRANSFORMS OF CROSS-CORRELATION FUNCTIONS
*       FILE NO. 149 ON THIS TAPE IS
*XSQDFR          SUM SQUARE DIF. OF FXD. VECTOR FROM ANOTHER OR FROM A CONSTANT
*       FILE NO. 150 ON THIS TAPE IS
*XSQRUT          SQUARE ROOT OF A FIXED VECTOR WITH ROUNDING
*       FILE NO. 151 ON THIS TAPE IS
*XVOVOV          DIVIDE ELEMENTS OF TWO FIXED VECTORS WITH OR WITHOUT ROUNDING
*       FILE NO. 152 ON THIS TAPE IS
*ZEFBCD          TEST IF NEXT TAPE RECORD IS END OF FILE AND REPOSITION TAPE
*       FILE NO. 153 ON THIS TAPE IS
*END TAPE CARD IN FORMAT(1H*,6X,8HEND TAPE)
```

3. Program Statistics

All of the programs of Set II are subroutines or functions, and the name of each program coincides with the name of the entry point to the subroutine or function. In the case of multiple-entry routines the name of the program coincides with that of the first entry card in the deck, and is called the "principal entry". The total count of principal and secondary entries is 395.

The program statistics tabulation which follows provides an alphabetical listing of all entries, with their secondary entries, transfer vectors, storage requirements, acceptance dates of symbolic deck, symbolic deck card counts, binary card counts, authors, and language. The symbol "M" is used for machine language (i.e. FAP), and "F" for FORTRAN. Authors are coded by initials as follows.

AMN	Arcadio M. Niell
CP	Cheh Pan
EAR	Enders A. Robinson
IH	Ira Hanson
JC	Jacqueline Clark
JFC	Jon F. Claerbout
JNG	James N. Galbraith, Jr.
JTO	J. T. Olsztyn
JTP	Joseph T. Procito, Jr.
MIT	MIT Lincoln Lab or Computation Center Staff
RAW	Ralph A. Wiggins
RJG	Roy J. Greenfield
SMS	Stephen M. Simpson, Jr.

ENTRY NAME	SECONDARY ENTRIES	TRANSFER VECTOR	STORAGE	DATE OF SYMBOLIC DECK	CARD SYMBOL COUNT	CARD BINARY COUNT	AUTHOR COUNT	LANGUAGE
ABSVAL			50	9/29/64	117	4	SMS	M
ADANL			183	9/29/64	336	11	JFC	M
	XDANL	SIN						
	ADANX							
	XDANX							
ADANX	(SEE ADANL)							
ADDK			114	9/29/64	366	8	SMS	M
	SUBK							
	MULK							
	DIVK							
	XADDK							
	XSUBK							
	XMULK							
	XDIVK							
	XDVRK							
	ADDKS							
	SUBKS							
	MULKS							
	DIVKS							
	XADDKS							
	XSUBKS							
	XMULKS							
	XDIVKS							
	XDVRKS							
ADDKS	(SEE ADDK)							
AMPHZ			149	10/ 1/64	251	10	JFC	M
	REIM	ATAN						
		SQRT						
		RND						
		COS						
		SIN						
ARBCOL			129	9/ 9/64	271	8	SMS	M
		INTOPR						

PROGRAM STATISTICS

ARCTAN TO CMPARP

ARCTAN	29	9/ 4/64	92	3	RAW	M
ATAN						
ARG (SEE LOCATE)						
ASPECT	278	9/29/64	536	15	SMS	M
COLAPS						
COSP						
DUBLX						
DUBLL						
SPLIT						
RVPRTS						
ASPEC2	74	3/15/65	206	5	SMS	M
SEQSAC						
NEXCOS						
AVRAGE	24	9/29/64	79	3	SMS	M
BLKSUM	49	9/ 4/64	169	4	SMS	M
BOOST	34	9/29/64	147	3	SMS	M
XBOOST						
DPRESS						
XDPRSS						
CALL (SEE LOCATE)						
CALL2 (SEE LOCATE)						
CARIGE	47	9/29/64	98	4	SMS	F
(STH)						
(FIL)						
CHISQR	105	9/29/64	85	6	JNG	F
CHOOSE	17	9/ 4/64	84	2	SMS	M
CHPRTS	76	9/29/64	149	5	SMS	M
RVPRTS						
CHSIGN	18	9/29/64	78	2	SMS	M
CHUSET (SEE INDEX)						
CLKON	46	9/29/64	42	4	RAW	F
CLOCK1						
(SPH)						
(FIL)						
CLOCK1 (7090)	57	3/15/65	148	4	SMS	M
CMPARL (SEE CMPARV)						
CMPARP	53	9/29/64	151	4	SMS	M
CMPARS						

PROGRAM STATISTICS

CMPARS TO COSISP

CMPARS (SEE CMPARP)							
CMPARV		50	9/ 4/64	156	4	SMS	M
CMPARL							
CMPRA		18	9/ 4/64	104	2	RAW	M
XCMPRA							
CMPRFL							
CMPRFL (SEE CMPRA)							
CNTROB		550	9/ 9/64	251	27	SMS	F
	SETVEC						
	LOG						
	CONTUR						
	EXP						
	SAME						
	(STH)						
	(FIL)						
CNTROW		802	9/ 9/64	521	39	SMS	F
	RNDDN						
	RNDUP						
	QUFIT1						
	CUFIT1						
	FASCUB						
	RND						
COLABL		185	9/ 4/64	124	10	SMS	F
	GENHOI						
	(SPH)						
	(FIL)						
	(STH)						
COLAPS		50	9/29/64	128	4	JC	M
CONTUR		587	9/ 9/64	642	29	SMS	F
	RNDDN						
	RNDUP						
	(STH)						
	(FIL)						
	COLABL						
	ARBCOL						
	CNTROW						
	SWITCH						
	(SPH)						
	XSAME						
CONVLV		96	9/29/64	99	6	JFC	F
CONVLV-II		56	10/ 2/64	149	4	JFC+ RAW	M
COSISP (SEE COSP)							

PROGRAM STATISTICS

* COSIS1 TO CVSOUT *

COSIS1		406	9/10/64	264	21	RAW	F
	IXCARG						
	SPLIT						
	MOVREV						
	CHPRTS						
	COSP						
	SISP						
	COSISP						
COSP		504	9/29/64	878	27	SMS	M
SISP							
COSISP							
COSTBL		121	9/29/64	200	8	JFC	M
SINTBL	COS						
COSTBX	SIN						
SINTBX							
COSTBX (SEE COSTBL)							
CPYFL2		178	9/ 9/64	304	10	RAW	M
	(IOS)						
	(TCO)						
	(WRS)						
	(RCH)						
	(TRC)						
	(ETT)						
	(WEF)						
	(BSR)						
	(RDS)						
CROSS		107	9/29/64	87	7	RAW	F
	STZ						
	FDOT						
CROST		134	9/29/64	99	8	RAW	F
	CROSS						
	REVERS						
CRSVM		327	9/10/64	220	17	RAW	F
	SETKS						
	MDOT3						
	STZ						
CSOUT		49	9/ 4/64	127	4	RAW	M
	CARIGE						
	(STH)						
	PRADJ						
	(FIL)						
CUFIT1		158	9/ 4/64	326	9	SMS	M
CVSOUT		84	9/29/64	221	6	SMS	M
	CARIGE						

PROGRAM STATISTICS

CVSOUT TO DUBLX

	FMTOUT						
	VECOUT						
DADECK		100	9/ 4/64	70	6	JNG+	F
	EOFSET					RAW	
	(TSH)						
	(RTN)						
	(STH)						
	(FIL)						
	RSKIP						
DELTA		17	9/ 4/64	141	2	SMS	M
XDELTA							
STEPR							
XSTEPR							
STEPL							
XSTEPL							
STEPC							
XSTEPC							
DERIVA		61	9/29/64	160	5	SMS	M
DETRM (SEE SIMEQ)							
DIFPRS		30	9/29/64	118	3	SMS	M
XDFPRS							
DISPLA (709)		220	9/29/64	474	12	MIT	M
	(IOH)						
DISPLA (7090)		219	9/ 4/64	481	13	MIT	M
	(IOH)						
	FRAME						
DIVIDE		23	9/29/64	88	3	SMS	M
DIVK (SEE ADDK)							
DIVKS (SEE ADDK)							
DO (SEE SEVRAL)							
DOTJ		59	10/ 2/64	143	4	RAW	M
DOTF		264	9/29/64	147	14	RAW	F
	DOTJ						
DPRESS (SEE BOOST)							
DSPFMT		194	9/29/64	313	11	SMS	M
DUBLL (SEE DUBLX)							
DUBLX		45	9/29/64	129	4	SMS	M
DUBLL							

HALVX						
HALVL						
ENDFIL (SEE REREAD)						
EOFSET (SEE REREAD)						
EXCHVS	22	9/29/64	84	3	SMS	M
EXPAND	189	9/ 4/64	380	11	SMS	M
INTOPR						
FACTOR	308	9/ 8/64	489	17	JNG	M
MAXAB						
LOG						
COSTBL						
COSP						
EXP						
FAPSUM	14	9/29/64	66	2	JFC	M
FASCN1	107	9/29/64	199	7	SMS	M
FASCOR (SEE PROCOR)						
FASCR1 (SEE PROCOR)						
FASCUB	141	9/ 4/64	260	9	SMS	M
FASEPC (SEE PROCOR)						
FASEP1 (SEE PROCOR)						
FASTRK	26	9/ 8/64	119	3	SMS	M
FDOT	40	9/ 4/64	101	3	RAW	M
FDOTR						
FDOTR (SEE FDOT)						
FIRE2	271	9/ 8/64	152	14	RAW	F
IXCARG						
STZ						
DOTP						
MATML3						
DOTJ						
FIXV	35	9/29/64	105	3	SMS	M
FIXVR						
FIXVR (SEE FIXV)						
FLDATA (SEE FXDATA)						
FLOATM	25	9/29/64	91	3	SMS	M

PROGRAM STATISTICS

FLOATV TO GNHOL2

FLOATV		22	9/29/64	81	3	SMS	M
FMTOUT		51	9/29/64	71	4	SMS	F
	FNDFMT						
	RPLFMT						
	(STH)						
	(FIL)						
FNDFMT		88	9/29/64	203	6	SMS	M
	REVER						
FRAME (709)		4	9/29/64	34	2	RAW	M
FRAME (7090)		9	9/ 4/64	47	2	MIT	M
FRQCT1		117	9/29/64	95	7	SMS	F
FRQCT2		117	9/29/64	212	7	JNG	M
FSKIP		50	9/ 4/64	104	4	JFC	M
	(IOS)						
	(RDS)						
	(BSR)						
	(TCO)						
	(TEF)						
	(TRC)						
FT24		777	9/29/64	848	40	CP	M
	FXDATA						
	FLDATA						
FT24 -II		818	9/29/64	147	39	RAW	F
FXDATA		102	10/ 1/64	248	7	SMS	M
FLDATA							
GENHOL		48	3/15/65	145	4	RAW	M
	(IOH)						
GETHOL		169	9/29/64	176	9	SMS	F
	XLOC						
	REVERS						
GETRD1		229	10/ 1/64	173	10	SMS	F
	(TSH)						
	(RTN)						
GETX		31	9/ 4/64	128	3	RAW	M
IGETX							
GNFLT1		232	9/29/64	164	12	SMS	F
	COS						
GNHOL2		74	9/29/64	158	5	RAW	M
	(IOH)						

	(FIL)						
GRAPH		1499	9/29/64	1103	72	SMS	F
	DISPLA						
	(SPH)						
	(FIL)						
	LINE						
	LOG						
	EXP(2						
	XFIXM						
	FLOATM						
	DSPFMT						
	FRAME						
	XLOC						
	MVBLOK						
	SCPSCL						
	HSTPLT						
GRAPHX		123	9/29/64	154	7	SMS	F
	GRAPH						
	FRAME						
GRUP2		201	10/ 1/64	141	11	JNG	F
HALVL (SEE DUBLX)							
HALVX (SEE DUBLX)							
HLADJ		46	9/29/64	111	4	SMS	M
HRADJ							
HRADJ (SEE HLADJ)							
HSTPLT		145	9/29/64	346	9	JNG	M
	LINEH						
	LINEV						
HSTPLT-II		188	9/29/64	336	11	RAW	M
	LINEH						
	LINEV						
HSTPLT-III (709)		256	9/29/64	438	14	RAW	M
	LINEH						
HSTPLT-III (7090)		258	9/ 8/64	446	14	RAW	M
	LINEH						
HVTOIV		39	9/29/64	110	3	SMS	M
IDERIV		54	9/29/64	149	4	SMS	M
**IF* (SEE SEVRAL)							
IFNCTN		208	9/ 4/64	444	12	SMS	M
	MONOCK						
	REVER						

PROGRAM STATISTICS

IGETX TO KIINT1

IGETX (SEE GETX)							
IINTGR		49	9/29/64	157	4	SMS	M
INDATA		896	10/ 1/64	489	32	UFC	F
	VARARG						
	FSKIP						
	(TSB)						
	(RLR)						
	FAPSUM						
	LOC						
	MVBLOK						
	XSAME						
	(SPH)						
	(FIL)						
	(STH)						
	UNPAKN						
INDEX		50	9/ 4/64	270	4	SMS	M
vINDEX							
SETEST							
SETAPT							
CHUSET							
INTGRA		47	9/29/64	175	4	SMS	M
INTHOL		72	9/ 9/64	156	5	RAW	M
	FNDFMT						
	(IOH)						
	(RTN)						
INTMSB (SEE TIMSUB)							
INTOPR		111	9/ 4/64	251	7	SMS	M
INTSUM		27	9/29/64	110	3	SMS	M
XNTSUM							
IPLYEV		98	10/ 2/64	84	6	RAW	F
	(IFMP)						
ITOMLI		37	9/29/64	98	3	SMS	M
IVTOHV		70	3/15/65	148	5	SMS	M
IXCARG		35	9/29/64	67	3	SMS	F
	XLGC						
KIINT1		191	9/29/64	129	10	SMS	F
	SQRT						
	EXP(3						
	NOINT1						

KOLAPS		100	9/29/64	219	6	JC	M
LIMITS		44	9/ 8/64	162	4	SMS	M
LINE	(709)	91	9/29/64	193	6	SMS	M
LINE	(7090)	95	9/ 4/64	208	6	SMS	M
LINEH	(709)	34	9/29/64	158	3	JNG	M
LINEH	(7090)	35	9/ 4/64	168	3	JNG	M
LINEV	(709)	34	9/29/64	161	3	JNG	M
LINEV	(7090)	35	9/ 4/64	169	3	JNG	M
LINIR1		96	9/29/64	93	6	SMS	F
LISTNG		755	9/29/64	221	38	RAW	F
	(RWT)						
	(STH)						
	(FIL)						
	(TSB)						
	(RLR)						
	FAPSUM						
	SAME						
	XSAME						
	(SPH)						
	FSKIP						
	SHFTR2						
LOC		4	9/29/64	54	2	RAW	M
LOCATE		512	3/15/65	2008	28	SMS	M
WHERE							
CALL							
CALL2							
SETSBV							
SETUP							
RETURN							
XINDEX							
ARG							
XARG							
STORE							
XNARGS							
XNAME							
LSHFT		12	9/29/64	72	2	RAW	M
XLSHFT							
LSLINE		117	10/ 1/64	82	7	RAW	F
LSSS1		122	9/29/64	116	7	RAW	F
	FDOT						

PROGRAM STATISTICS

MATINV TO MINSNM

MATINV		90	9/29/64	79	6	RAW	F
	SIMEQ						
MATML1		61	9/29/64	137	5	RAW	M
MATML3		120	9/29/64	105	7	RAW	F
	DOTJ						
MATRA		92	9/29/64	177	5	RAW+SMS	M
MATRA1		42	9/29/64	95	4	RAW	M
MAXAB	(SEE MAXSN)						
MAXABM	(SEE MAXSNM)						
MAXSN		54	9/29/64	170	5	JFC	M
MINSN							
MAXAB							
MINAB							
MAXSNM		61	9/ 4/64	247	5	SMS	M
MINSNM							
MAXABM							
MINABM							
MDOT		109	9/29/64	94	7	RAW	F
	MATML1						
MDOT3		122	9/29/64	120	7	RAW	F
	MATML3						
MEMUSE		71	9/ 4/64	69	5	SMS	F
	XLCOMN						
	(STH)						
	(FIL)						
MFACT		187	9/29/64	103	10	RAW	F
	STZ						
	DOTJ						
	SQRT						
MIFLS		276	9/ 8/64	167	14	RAW	F
	MOVREV						
	MATML3						
MINAB	(SEE MAXSN)						
MINABM	(SEE MAXSNM)						
MINSN	(SEE MAXSN)						
MINSNM	(SEE MAXSNM)						

MIPLS		571	9/29/64	254	28	RAW	F
	IXCARG						
	MATINV						
	MATML3						
	MATRA						
	MDOT3						
	MOVREV						
	STZ						
MISS		335	10/ 5/64	150	17	RAW	F
	MOVREV						
	MATML3						
	MDOT3						
MLISCL		47	9/29/64	115	4	SMS	M
MLI2A6		128	9/29/64	218	8	SMS	M
MONOCK		48	9/ 4/64	165	4	SMS	M
MOUT		130	9/ 8/64	101	8	RAW	F
	CARIGE						
	(STH)						
	(FIL)						
MOUTAI		357	9/ 4/64	295	18	SMS	F
	EXP(2						
	CARIGE						
	GNHOL2						
	MAXABM						
	LOG						
	RND						
	(STH)						
	(FIL)						
	SAME						
	MOVE						
	MULPLY						
	FIXVR						
MOVE		32	9/29/64	92	3	JFC	M
MOVECS		24	9/29/64	106	3	SMS	M
	MOVE						
MOVREV		74	9/29/64	156	5	RAW	M
MPSEQ1		110	9/29/64	197	7	JNG	M
MRVRS		61	9/29/64	67	4	RAW	F
	REVERS						
MSCON1		238	9/29/64	108	11	JNG	F
MULK (SEE ADDK)							

MULK -II		76	9/29/64	78	5	SMS	F
	SETUP						
	ARG						
	STORE						
	RETURN						
MULKS (SEE ADDK)							
MULLER		757	9/ 9/64	232	36	IH	F
	SQRT						
MULPLY		34	9/29/64	114	3	SMS	M
XMLPLY							
MUVADD		129	9/29/64	245	8	SMS	M
MVBLOK		19	9/29/64	83	2	SMS	M
MVINAV		221	9/29/64	116	12	SMS	F
MVNSUM		71	9/ 4/64	202	5	SMS	M
MVNTIN		88	9/ 4/64	234	6	SMS	M
MVNTNA							
MVNTNA (SEE MVNTIN)							
MVSQAV		236	9/29/64	116	13	SMS	F
MYRARE		302	9/29/64	250	16	SMS	F
	EXP(2						
NEXCOS (SEE SEQSAC)							
NEXSIN (SEE SEQSAC)							
NMZMG1		34	9/29/64	97	3	RAW	M
NOINT1		369	9/29/64	375	20	SMS+	M
NOINT2	LINTR1					JNG	
NOINT2 (SEE NOINT1)							
NRMVEC		112	9/29/64	100	7	RAW	F
	SQRT						
	MAXAB						
NTHA		11	10/ 6/64	93	2	SMS	M
XNTHA							
NURINC		121	9/ 4/64	327	8	SMS	M
NXALRM		243	9/29/64	178	13	SMS	F
	FASCN1						

PROGRAM STATISTICS

****
* ONLINE TO PLOTVS *
****

ONLINE		134	4/14/65	191	8	RAW	M
(STH)	(IOH)						
(STHM)	(WER)						
(STHD)	(TES)						
	(WRS)						
	(WTC)						
	(RCH)						
	(FIL)						
	(SPH)						
OUDATA		495	3/15/65	269	11	IFC	F
	VARARG						
	LOC						
	MVBLOK						
	FAPSUM						
	PAKN						
	(STB)						
	(WLR)						
	(EFT)						
PACDAT		152	9/ 9/64	259	9	RAW	M
	(IOS)						
	(TCO)						
	(RDS)						
	(RCH)						
	(ETT)						
PAKN		78	9/29/64	147	5	JFC	M
	FXDATA						
PLANSP		1169	9/ 9/64	383	56	RAW	F
	SETKS						
	LIMITS						
	IXCARG						
	CHOOSE						
	XOOZE						
	MOVREV						
	STZ						
	ROAR2						
	XADDKS						
	KOLAPS						
	COSTBL						
	SINTBL						
	XADDK						
	COSIS1						
	MATRA						
PLOTVS		494	9/ 4/64	261	18	SMS	F
	SETVEC						
	SETKS						
	SETKV						
	(STH)						
	(FIL)						
	SWITCH						
	(SPH)						
	RND						

PROGRAM STATISTICS

PLTVS1 TO PROCOR

PLTVS1		817	9/ 4/64	393	40	SMS	F
	VARARG						
	SETKS						
	SETVEC						
	SETKVS						
	XSTLIN						
	XLOC						
	XSAME						
	RMSDEV						
	(STH)						
	(FIL)						
	MAXSN						
	MINSN						
	MULPLY						
	BOOST						
	PLOTVS						
	OPRESS						
PLURAL (SEE SEVRAL)							
PLURNS		73	9/29/64	247	5	SMS	M
PLYSYN		172	10/ 5/64	162	10	EAR	F
	COS						
	CONVLV						
POKCT1		219	9/29/64	134	11	SMS	F
	FRQCT1						
POLYDV		130	9/ 9/64	102	7	JFC+	F
	MOVE					RAW	
	STZ						
POLYEV		54	9/29/64	62	4	JFC	F
POLYSN		256	9/ 8/64	167	14	RAW	F
	SQRT						
	COS						
	CONVLV						
	MOVE						
POWER		50	9/29/64	130	4	SMS	M
SMPRDV	EXP(2						
PRBFIT		373	9/29/64	187	16	RJG	F
	SQRT						
	EXP(2						
	FXP						
PROB2		229	10/ 6/64	175	12	JNG	F
PROCOR		770	9/29/64	1499	40	SMS	M
FASCOR							
FASEPC							

PRCCOR TO QXCOR1 | PROCOR TO QXCOR1

FASCR1							
FASEP1							
PSQRT		155	10/ 5/64	91	9	JFC	F
	SQRT						
PWMLIV		300	9/29/64	142	15	SMS	F
	ML12A6						
	(STH)						
	(FIL)						
	(SPH)						
QACORR		207	9/29/64	184	11	SMS	F
	FXDATA						
	PROCOR						
	FASCOR						
	FLDATA						
QCNVLV		569	9/29/64	294	27	SMS	F
	XLOC						
	FXDATA						
	PROCOR						
	FASCOR						
	FASEPC						
	FLDATA						
QFURRY		244	9/29/64	181	13	SMS	F
	ST2						
	MOVE						
	COSTBL						
	SINTBL						
	XSPECT						
QIFURY		280	9/29/64	206	14	SMS	F
	COSTBL						
	SINTBL						
	COSISP						
	XLOC						
QINTR1		229	9/ 4/64	192	12	JTP	F
	RNDUP						
	QUFIT1						
QUFIT1		79	9/ 4/64	200	5	SMS	M
QXCORR		283	9/29/64	249	15	SMS	F
	XLOC						
	FXDATA						
	PROCOR						
	FASCOR						
	FLDATA						
QXCOR1		502	3/15/65	198	25	RAW	F
	SETKS						
	IXCARG						

PROGRAM STATISTICS

• QXCOR1 TO RLSPR2 •

	LIMITS						
	STZ						
	REVERS						
	PROCOR						
	FASCR1						
	FASEP1						
RDATA		645	3/15/65	396	31	RAW	F
	SETUP						
	RETURN						
	IXCARG						
	(TSH)						
	(RTN)						
	(STH)						
	(FIL)						
	HVTOIV						
	IVTOHV						
	CMPRA						
	ARG						
	INTHOL						
	STORE						
REFIT (SEE SPLIT)							
REFLEC		28	9/29/64	108	3	SMS	M
XRFLEC							
REIM (SEE AMPHZ)							
REMAV		36	9/29/64	106	3	SMS	M
REREAD		114	9/ 9/64	283	7	RAW	M
EOFSET	(IOH)						
ENDFIL	(RDS)						
(TSH)	(RDC)						
(TSHM)	(RCH)						
	(TCO)						
	(TEF)						
	EXIT						
	(RER)						
RETURN (SEE LOCATE)							
[illegible]ER		30	9/29/64	98	3	SMS	M
REVERS		29	9/29/64	77	3	RAW	M
RLSPR		142	10/ 5/64	121	8	RAW	F
	FOOTR						
RLSPR2		700	9/ 9/64	281	34	RAW	F
	IXCARG						
	STZ						
	MOVREV						
	DOTP						

	MATML3						
	DOTJ						
	SIMEQ						
RLSSR		82	9/29/64	115	5	RAW	F
	FDOTR						
RMSDAV (SEE RMSDEV)							
RMSDEV		50	9/ 4/64	160	4	SMS	M
RMSDAV	SQRT						
RND		15	9/29/64	79	2	RAW	M
RNDUP							
RNDDN							
RNDDN (SEE RND)							
RNDUP (SEE RND)							
RNDV		34	9/29/64	118	3	SMS	M
RNDVUP	RND						
RNDVDN	RNDUP						
	RNDDN						
RNDVDN (SEE RNDV)							
RNDVUP (SEE RNDV)							
ROAR2		174	9/10/64	114	9	RAW	F
	MATRA						
	MOVREV						
	REVERS						
ROTAT1		46	9/ 4/64	110	4	RAW+ JC	M
RPLFMT		17	9/29/64	85	2	SMS	M
RSKIP		37	9/29/64	90	3	RAW	M
	(IOS)						
	(TRC)						
	(TCO)						
	(TEF)						
	(RDS)						
	(BSR)						
RVPRTS (SEE CHPRTS)							
SAME		1	9/29/64	40	2	JFC	M
XSAME							
SCPSCL		33	9/29/64	111	3	SMS	M
SEARCH		25	9/29/64	95	3	RAW	M

PROGRAM STATISTICS

SEQSAC TO SETVEC

SEQSAC		94	9/ 8/64	278	6	SMS	M
NEXCOS	COS						
NEXSIN	SIN						
SETAPT (SEE INDEX)							
SETEST (SEE INDEX)							
SETINO		84	9/ 8/64	92	6	SMS	F
	XLIMIT						
	(RWT)						
	(TSB)						
	(RLR)						
	FSKIP						
SETK		37	9/29/64	190	3	SMS	M
SETKS							
SETVEC							
SETK -II		63	9/29/64	73	4	SMS	F
	SETUP						
	STORE						
	RETURN						
SETKP		40	9/29/64	124	3	SMS	M
SETVCP	SETK						
	SETVEC						
SETKS (SEE SETK)							
SETKS -II		91	9/29/64	86	6	SMS	F
	SETUP						
	ARG						
	STORE						
	RETURN						
SETKV		15	9/29/64	75	2	SMS	M
SETKVS		25	9/29/64	106	3	SMS	M
SETLIN		27	9/29/64	95	3	SMS	M
XSTLIN							
SETLNS		39	9/29/64	124	3	SMS	M
	SETLIN						
	XSTLIN						
SETSBV (SEE LOCATE)							
SETUP (SEE LOCATE)							
SETVCP (SEE SETKP)							
SETVEC (SEE SETK)							

SEVRAL		416	9/29/64	949	22	SMS	M
PLURAL	LOCATE						
''DO''	WHERE						
''IF''							
SHFTR1		70	9/29/64	158	5	SMS	M
SHFTR2		72	9/29/64	163	5	SMS+	M
						RAW	
SHUFFL		101	9/ 8/64	125	6	SMS	F
	GETRD1						
	SEARCH						
	SIZEUP						
SIFT		30	9/ 4/64	118	3	SMS	M
SIMEQ		441	9/ 9/64	642	24	JTO+	M
DETRM						AMN+	
						RAW	
SINTBL (SEE COSTBL)							
SINTBX (SEE COSTBL)							
SISP (SEE COSP)							
SIZEUP		136	3/15/65	247	8	RAW+	M
SIZUPL						SMS	
SIZUPL (SEE SIZEUP)							
SMPRDV (SEE POWER)							
SMPSON		317	9/ 4/64	197	17	JNG	F
SPCOR2		291	9/ 8/64	181	15	RAW	F
	XLOC						
	STZ						
	FXDATA						
	QXCOR1						
	FLDATA						
SPLIT		224	9/29/64	395	13	SMS	M
REFIT							
SQRDEV (SEE SQRDFR)							
SQRDFR		36	9/29/64	111	3	SMS	M
SQRDEV							
SQRMLI		55	9/29/64	128	4	SMS	M
SQROOT		24	9/29/64	83	3	SMS	M
	SQRT						

PROGRAM STATISTICS

SQRSUM TO TINGL

SQRSUM		36	9/29/64	107	3	SMS	M
XSQSUM							
SQUARE		32	9/29/64	111	3	SMS	M
XSQUAR							
SRCH1		93	9/ 8/64	93	6	RAW	F
	XACTEQ						
STEPC	(SEE DELTA)						
STEPL	(SEE DELTA)						
STEPR	(SEE DELTA)						
(STH)	(SEE ONLINE)						
(STHD)	(SEE ONLINE)						
(STHM)	(SEE ONLINE)						
STORE	(SEE LOCATE)						
STZ		14	9/29/64	60	2	JFC	M
STZS		24	9/29/64	97	3	SMS	M
SUBK	(SEE ADDK)						
SUBKS	(SEE ADDK)						
SUM		23	9/29/64	92	3	SMS	M
XSUM							
SUMDEV	(SEE SUMDFR)						
SUMDFR		44	9/29/64	156	4	SMS	M
XSMDFR							
SUMDEV							
XSMDEV							
SWITCH		15	9/ 4/64	84	2	SMS	M
TAMVL		63	9/ 4/64	189	5	SMS	M
TAMVR							
TAMVR	(SEE TAMVL)						
TIMA2B		124	9/ 9/64	258	8	SMS+RAW	M
TIMSUB		229	9/ 8/64	450	13	SMS+RAW	M
INTMSB	TIMA2B						
TINGL		43	9/ 8/64	147	4	SMS	M
TINGLA							

PROGRAM STATISTICS

TINGLA TO WAC

TINGLA (SEE TINGL)							
TRMINO		67	9/ 4/64	77	5	SMS	F
	XLIMIT						
	OUDATA						
	FSKIP						
	(RWT)						
(TSH) (SEE REREAD)							
(TSHM) (SEE REREAD)							
UNPAKN		78	9/ 9/64	150	5	JFC	M
VARARG		44	9/29/64	132	4	JFC	M
VDOTV		25	9/ 4/64	121	3	SMS	M
VDVBYV		22	9/29/64	90	3	SMS	M
VECOUT		66	9/29/64	91	5	SMS	F
	FMDFMT						
	RPLFMT						
	(STH)						
	(FIL)						
VINDEX (SEE INDEX)							
VMNUSV (SEE VPLUSV)							
VOUT		104	9/29/64	111	7	SMS	F
	CARIGE						
	HRADJ						
	(STH)						
	(FIL)						
	VECOUT						
VPLUSV		34	9/29/64	127	3	SMS	M
XVPLSV							
VMNUSV							
XVMNSV							
VRSOUT		47	9/29/64	138	4	SMS	M
	CARIGE						
	VECOUT						
VSOUT		37	9/29/64	125	3	SMS	M
	VOUT						
VTIMSV		34	9/29/64	112	3	SMS	M
XVTMSV							
WAC		107	9/29/64	83	6	JFC	F

PROGRAM STATISTICS

WHERE TO XDPRSS

WHERE (SEE LOCATE)							
WHICH		4	9/ 4/64	77	2	SMS	M
XWHICH							
WLLSFP		217	10/ 6/64	264	11	RAW	F
	FDOTR						
	FDOT						
	MOVE						
WRTDAT		77	9/ 8/64	126	5	RAW	M
	(IOS)						
	(TCO)						
	(WRS)						
	(RCH)						
	(TRC)						
	(ETT)						
XACTEQ		11	9/ 4/64	76	2	SMS	M
XADDK (SEE ADDK)							
XADDKS (SEE ADDK)							
XARG (SEE LOCATE)							
XAVRGE		34	9/29/64	104	3	SMS	M
XAVRGR	XDIV						
	XDIVR						
XAVRGR (SEE XAVRGE)							
XBOOST (SEE BOOST)							
XCMPRA (SEE CMPRA)							
XDANL (SEE ADANL)							
XDANX (SEE ADANL)							
XDELTA (SEE DELTA)							
XDFPRS (SEE DIFPRS)							
XDIV		27	9/29/64	109	3	SMS	M
XDIVR							
XDIVK (SEE ADDK)							
XDIVKS (SEE ADDK)							
XDIVR (SEE XDIV)							
XDPRSS (SEE BOOST)							

XDVIDE		33	9/29/64	105	3	SMS	M
XDVIDR	XDIV						
	XDIVR						
XDVIDR (SEE XDVIDE)							
XDVRK (SEE ADDK)							
XDVRKS (SEE ADDK)							
XFIXM		31	9/29/64	98	3	SMS	M
XINDEX (SEE LOCATE)							
XLCOMN		14	9/ 4/64	76	2	RAW	M
XLIMIT		25	9/ 4/64	101	3	SMS	M
XLOCV		24	9/ 4/64	100	3	SMS	M
XLSHFT (SEE LSHFT)							
XMLPLY (SEE MULPLY)							
XMULK (SEE ADDK)							
XMULKS (SEE ADDK)							
XNAME (SEE LOCATE)							
XNARGS (SEE LOCATE)							
XNTHA (SEE NTHA)							
XNTSUM (SEE INTSUM)							
XOOZE		4	9/ 4/64	61	2	SMS	M
XREMAV		31	9/29/64	112	3	SMS	M
	XAVRGR						
XRFLEC (SEE REFLEC)							
XSAME (SEE SAME)							
XSMDEV (SEE SUMDFR)							
XSMDFR (SEE SUMDFR)							
XSPECT		523	9/29/64	239	26	SMS	F
	SPLIT						
	COSISP						
	REFIT						
	XLOC						
	KOLAPS						
	CHPRIS						

XSQDEV (SEE XSQDFR)							
XSQDFR		37	9/29/64	113	3	SMS	M
XSQDEV							
XSQRUT		37	9/29/64	103	3	SMS	M
	FIXVR						
	SQRT						
XSQSUM (SEE SQRSUM)							
XSQUAR (SEE SQUARE)							
XSTEPC (SEE DELTA)							
XSTEPL (SEE DELTA)							
XSTEPR (SEE DELTA)							
XSTLIN (SEE SETLIN)							
XSUBK (SEE ADDK)							
XSUBKS (SEE ADDK)							
XSUM (SEE SUM)							
XVDRBV (SEE XVDVBV)							
XVDVBV		34	9/29/64	109	3	SMS	M
XVDRBV	XDIV						
	XDIVR						
XVMNSV (SEE VPLUSV)							
XVPLSV (SEE VPLUSV)							
XVTMSV (SEE VTIMSV)							
XWHICH (SEE WHICH)							
ZEFBCD		54	9/ 8/64	129	4	JNG	M
ZEFBIN	(IOS)						
	(RDS)						
	(RCH)						
	(TCO)						
	(TEF)						
	(TPC)						
	(BSR)						
ZEFBIN (SEE ZEFBCD)							

4. Conventions Used in Program Writeups

The general format of preparation of symbolic decks we have adhered to is illustrated by the sample listings shown on the next few pages for the two very short routines CONVLV and RND (File 28 of Tape 1 and File 79 of Tape 2). In all cases the general sequence is 1) Control Cards, 2) Subroutine or Entry cards, 3) Comment cards giving Abstract (including language, equipment, length, speed, and author), 4) Comment cards giving Usage (including FORTRAN usage, transfer vector, input-output descriptions, and examples), and 5) Program proper. All cards are serialized after the first one, in columns 76-79. The following observations should assist the interpretation of our comment cards.

1. All programs are designed to operate under the FORTRAN-II system.

2. In general we adhere to FORTRAN conventions in naming fixed, floating point, octal, and hollerith variables regardless of whether the program is FAP or FORTRAN. This convention should always be assumed for subroutine arguments unless otherwise noted.

3. The term "FORTRAN INTEGER" or FORTRAN-II INTEGER" or sometimes just "INTEGER" is used to refer to a fixed point integer in the decrement (binary point between bits 17 and 18,

```
*     CONVLV (SUBROUTINE)                    9/29/64   LAST CARD IN DECK IS NO. 0098
*     LABEL                                                                    0001
CCONVLV                                                                        0002
      SUBROUTINE CONVLV(LX,XX,LY,YY,CC)                                        0003
C                                                                              0004
C                    ----ABSTRACT----                                          0005
C                                                                              0006
C TITLE - CONVLV                                                               0007
C       COMPLETE CONVOLUTION OF TWO TRANSIENTS                                 0008
C                                                                              0009
C             CONVLV CONVOLVES TWO TRANSIENTS, X(I) I=0,1,...,LX-1             0010
C             AND Y(I) I=0,1,...,LY-1 , TO PRODUCE THE COMPLETE                0011
C             CONVOLUTION FUNCTION                                             0012
C                                                                              0013
C                                LX-1                                          0014
C                      C(I)  =  SUM ( X(J)*Y(I-J) )                            0015
C                                J=0                                           0016
C                                                                              0017
C                      FOR I = 0,1,...,LX+LY-2                                 0018
C                      WHERE                                                   0019
C                         LX AND LY ARE INPUT PARAMETERS                       0020
C                         Y(K) IS ASSUMED = 0.0 FOR K OUTSIDE OF               0021
C                              THE RANGE 0 TO LY-1                             0022
C             NOTE THAT THE CONVOLUTION IS INDEPENDENT OF THE ORDER            0023
C             OF THE INPUTS X AND Y.                                           0024
C                                                                              0025
C             TECHNIQUE USED IS AN ALGORITHM BASED ON ANALOGY TO               0026
C             MULTIPLICATION OF POLYNOMIALS                                    0027
C                                                                              0028
C LANGUAGE  - FORTRAN II SUBROUTINE                                            0029
C EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)                                    0030
C STORAGE   - 96  REGISTERS                                                    0031
C SPEED     - ABOUT .49 * (LX*LY)  MILLISEC ON THE 709                         0032
C             ABOUT .082 * (LX*LY)  MILLISEC ON THE 7090                       0033
C AUTHOR    - J. CLAERBOUT                                                     0034
C                                                                              0035
C                    ----USAGE----                                             0036
C                                                                              0037
C TRANSFER VECTOR CONTAINS ROUITNES - (NONE)                                   0038
(       AND FORTRAN SYSTEM ROUTINES - (NONE)                                   0039
C                                                                              0040
C FORTRAN USAGE                                                                0041
C     CALL CONVLV(LX,XX LY,YY,CC)                                              0042
C                                                                              0043
C INPUTS                                                                       0044
C                                                                              0045
C    LX         IS NO. OF TERMS IN X VECTOR                                    0046
C               MUST EXCEED ZERO (PROGRAM EXITS IF ZERO OR LESS)               0047
C                                                                              0048
C    XX(I)      I=1,...,LX  CONTAINS  X(0),...,X(LX-1) RESPECTIVELY            0049
C                                                                              0050
```

```
C     LY          IS NO. OF TERMS IN Y VECTOR                                    0051
C                 MUST EXCEED ZERO  (PROGRAM EXITS IF ZERO OR LESS)              0052
C                                                                                0053
C     YY(I)       I=1...LY    CONTAINS  Y(0),...,Y(LY-1) RESPECTIVELY            0054
C                 EQUIVALENCE (XX,YY  IS PERMITTED                               0055
C                                                                                0056
C OUTPUTS                                                                        0057
C                                                                                0058
C     CC(I)       I=1,...,LX+LY-1 CONTAINS C(0),...,C(LX+LY-2) RESPECTIVELY      0059
C                           WHERE C(I) IS GIVEN IN ABSTRACT                      0060
C                                                                                0061
C EXAMPLES                                                                       0062
C                                                                                0063
C 1. SHOWING REVERSIBILITY OF X AND Y                                            0064
C     INPUTS  - LX = 3  XX(1...3) = 1.,2.,3.                                     0065
C               LY = 2  YY(1...2) =  10.,1.                                      0066
C                                                                                0067
C USAGE   -         CALL CONVLV(LX,XX,LY,YY,CC1)                                 0068
C                   CALL CONVLV(LY,YY,LX,XX,CC2)                                 0069
C     OUTPUTS - CC1(1...4) = CC2(1...4) = 10.,21.,32.,3.                         0070
C                                                                                0071
C 2. ILLEGAL INPUT CASES (NO OUTPUT)                                             0072
C     INPUTS  - SAME AS EXAMPLE 1. EXCEPT START WITH OUTPUT VECTORS              0073
C                 CLEANED, I.E. CC1(1...4) = CC2(1...4) = 0.,0.,0.,0.            0074
C     USAGE   -       CALL CONVLV(-2,XX,LY,YY,CC1)                               0075
C                     CALL CONVLV(LX,XX,0,YY,CC2)                                0076
C     OUTPUTS - CC1(1...4) = 0.,0.,0.,0.   (ILLEGAL LX)                          0077
C               CC2(1...4) = 0.,0.,0.,0.   (ILLEGAL LY)                          0078
C                                                                                0079
C PROGRAM FOLLOWS BELOW                                                          0080
C                                                                                0081
C DUMMY DIMENSION STATEMENTS                                                     0082
      DIMENSION  XX(2),YY(2),CC(2)                                               0083
C CHECK LEGALITIES                                                               0084
      IF (LX)  9999,9999,10                                                      0085
   10 IF (LY)  9999,9999,20                                                      0086
C CLEAR OUTPUT VECTOR                                                            0087
   20 LC=LX+LY-1                                                                 0088
      DO 30  I=1,LC                                                              0089
   30 CC(I)=0.0                                                                  0090
C CONVOLVE                                                                       0091
      DO 40  I=1,LX                                                              0092
      DO 40  J=1,LY                                                              0093
      K=I+J                                                                      0094
   40 CC(K-1)=CC(K-1)+XX(I)*YY(J)                                                0095
C EXIT                                                                           0096
 9999 RETURN                                                                     0097
      END                                                                        0098
```

```
*       RND (FUNCTION)                    9/29/64   LAST CARD IN DECK IS NO. 0078
*       FAP                                                                  0001
*RND                                                                         0002
        COUNT   60                                                           0003
        LBL     RND                                                          0004
        ENTRY   RND     F(Y)                                                 0005
        ENTRY   RNDUP   F(Y)                                                 0006
        ENTRY   RNDDN   F(Y)                                                 0007
*                                                                            0008
*                    ----ABSTRACT----                                        0009
*                                                                            0010
* TITLE - RND    , WITH SECONDARY ENTRY POINTS RNDUP, RNDDN                   0011
*       ROUNDS FLTG. PT. NO. UP, DOWN, OR TO NEAREST FLTG. PT. INTEGER       0012
*                                                                            0013
*             RND ROUNDS A FLOATING POINT NUMBER TO THE NEAREST FLOATING     0014
*               POINT INTEGER.                                               0015
*                                                                            0016
*             RNDUP ROUNDS A POSITIVE (NEGATIVE) FLOATING POINT NUMBER       0017
*               TO THE NEXT HIGHER (LOWER) FLOATING POINT INTEGER.           0018
*                                                                            0019
*             RNDDN ROUNDS A POSITIVE (NEGATIVE) FLOATING POINT NUMBER       0020
*               TO THE NEXT LOWER (HIGHER) FLOATING POINT INTEGER.           0021
*                                                                            0022
* LANGUAGE  - FAP, FORTRAN II FUNCTION                                       0023
* EQUIPMENT - 709 OR 7090 (MAIN FRAME ONLY)                                  0024
* STORAGE   - 15 REGISTERS                                                   0025
* SPEED     - 26 MACHINE CYCLES FOR RND                                      0026
* AUTHOR    - R.A. WIGGINS,  15/9/62                                         0027
*                                                                            0028
*                    ----USAGE----                                           0029
*                                                                            0030
* TRANSFER VECTOR CONTAINS ROUTINES -   NONE                                 0031
*       AND FORTRAN SYSTEM ROUTINES -   NONE                                 0032
*                                                                            0033
* FORTRAN USAGE                                                              0034
*    X1 =   RNDF(Y)                                                          0035
*    X2 = RNDUPF(Y)                                                          0036
*    X3 = RNDDNF(Y)                                                          0037
*                                                                            0038
* INPUTS                                                                     0039
*                                                                            0040
*    Y          IS A FLOATING POINT NUMBER                                   0041
*               MUST BE LSTHN= 10.**9                                        0042
*                                                                            0043
* OUTPUTS                                                                    0044
*                                                                            0045
*    X1         IS A FLOATING POINT INTEGER                                  0046
*                                                                            0047
*    X2         IS A FLOATING POINT INTEGER                                  0048
*                                                                            0049
*    X3         IS A FLOATING POINT INTEGER                                  0050
```

```
*                                                                    0051
* EXAMPLES                                                           0052
*                                                                    0053
* 1. INPUT    - Y=104.2                                             0054
*    OUTPUTS  - X1=104.   X2=105.   X3=104.                         0055
*                                                                    0056
* 2. INPUT    - Y=.5                                                0057
*    OUTPUTS  - X1=1.     X2=1.     X3=0.                           0058
*                                                                    0059
* 3. INPUT    - Y=-49.7                                             0060
*    OUTPUTS  - X1=-50.   X2=-50.   X3=-49.                         0061
*                                                                    0062
* 4. INPUT    - Y=1015.                                             0063
*    OUTPUTS  - X1=1015.  X2=1015.  X3=1015.                        0064
*                                                                    0065
        BCI     1,RND                                               0066
RNDUP   TMI     A                                                   0067
        FAD     =0177777777777                                      0068
        FAD     =.5                                                 0069
RNDDN   UFA     =0233000000000                                      0070
        FAD     =0233000000000                                      0071
        TRA     1,4                                                 0072
A       FSB     =0177777777777                                      0073
        FSB     =.5                                                 0074
        TRA     RNDDN                                               0075
RND     TMI     A+1                                                 0076
        TRA     RNDUP+2                                             0077
        END                                                         0078
```

maximum magnitude = $2^{17}-1$).

4. The term "MACHINE LANGUAGE INTEGER" or "MACHINE INTEGER", or sometimes "MLI" is used to refer to fixed point integers in the address (binary point beyond bit 35, maximum magnitude = $2^{35}-1$).

5. The terms "LSTHN" and "LSTHN=" are equivalent to "<" and "≤". The terms "GRTHN" and "GRTHN=" are equivalent to ">" and "≥".

6. The names of all our subprogram-type routines (subroutines, functions) are always the same as their entry point (in the case of multiple entry point routines the first entry point listed is equated with the name). A serial number "-II" or "-III" following the name indicates that this program is one of a series, all of which have identical calling sequences and essentially the same functions, but the user must choose the appropriate one in terms of his requirements. A "(709)" following the name indicates that this routine can only be used on the 709. A "(7090)" indicates the program works on either the 7090 or the 7094. All the routines without such specification can be used on any of the three machines.

7. Expressions appearing under "ABSTRACT" may deviate from FORTRAN conventions. The emphasis here has been to produce expressions which are visually

close to those of ordinary mathematics.

8. In the listings of required routines as found in the transfer vectors we list separately the FORTRAN system routines (which can be ignored) and non-FORTRAN-system routines (which cannot be ignored). All of the non-FORTRAN system routines required are included somewhere in the program set. In this connection the word "NONE" or "(NONE)" means "none required" and does not refer to routines by those names.

9. It should be stressed that the transfer vector as listed is only the first level of subprogram requirements and the subprograms listed should be checked for further subprogram requirements. The table in Section 3 is probably the most rapid and accurate for determining the complete requirements.

10. In the usage of these programs it should be assummed that none of the subprogram arguments can be safely equated (either by equivalence statements or repeated use of the same name) except as specifically noted.

11. The numerical examples given involve some notation conventions which should be fairly obvious. Thus

    A) "IX(1...5) = 2,4,6,8,10" or
    "IX(1,2,...,5) = 2,4,6,8,10" stands for
    "IX(1) = 2", "IX(2) = 4," etc.

B) "OCT" stands for octal data

C) "MLI" is machine language integer

The representation of hollerith data is not too satisfactory or consistent as given here. In most cases we use either

X(1...) = 6H(something)
= 6Hsomething

to imply that the "something" is a string of hollerith characters stored 6 to a register (i.e. FORMAT(A6)). However, in some cases the "something" may be split into groups of six characters separated by commas to conform to a representation such as A) above. The reader will have to use his judgment from the context.

12. In the examples, if no "USAGE" is given, the user is to assume that, following the setting up of the "INPUTS", a "CALL" statement is to be executed in the exact literal form as given under "FORTRAN USAGE".

13. In the case of programs with scope output, blank comment cards are inserted at appropriate places in the example outputs so that photographs of the actual outputs can be pasted there on the listings.

14. Instructions equivalent to the linkage director have been inserted in many of the FAP programs so that they may operate properly with systems which do not have the standard error procedure. The pro-

grams will, of course, operate with systems which do have the standard error procedure option operative but the error tracing scheme will not be able to function completely since index register four will be stored in the "artificial linkage" director rather than in the one constructed by the assembler. In many cases the error procedure may be made completely operative by removing the PZE 0 and BCI 1, NAME cards appearing at the beginning of the program.

5. Magnetic Tape Copies

The following steps have been taken in the production of the master tape from which copies will be made.

1. All programs to be included had special test programs written which tested, among other things, all examples given in the program comment cards. These tests were passed individually.
2. The symbolic decks were divided into groups, each group being loaded on a separate tape.
3. Each such tape was then serialized and dated by a special program and then the serialized tapes were compiled to produce sets of binary decks.
4. The binary decks thus compiled were rerun through the test programs, and the test results compared with earlier test results.
5. The serialized tapes were merged by program to form the master tape.
6. The master tape was then compiled and the binaries from this compilations compared by the 519 reproducing punch against the binary decks used in step 4.

VELA UNIFORM associates desiring a copy of these programs should write their request to

Headquarters, USAF/AFTAC
VELA Seismological Center
Washington 25, D. C. 20330
ATTN: Major J. J. Connor

The letter should request

"MIT Geophysics Program Set II".

By separate mail the requester should also send two 2400' blank tapes.

6. KWIC Index to Programs

The remaining pages are a KWIC (Key Work in Context) index of the 267 programs in the program set (produced by the routine ROKWIC). Our coding in this index is as follows

Column	65	F means FORTRAN program
		M means FAP program
	66	Blank means FORTRAN-type subroutine or functions
		* means main program
	67-80	give the program name

KWIC Index

```
                          $FIND AVERAGE OF FIXED PT VECTOR      M      XAVRGE
T END                  $TRIANGULAR AVERAGING, MOVING LEFT OR RIGH M      TAMVL
GES                  $DIVIDE THE X AXIS INTO EQUALLY PROBABLE RAN F      GRUP2
                   $SKIP FORWARD OR BACKWARD OVER FILES ON TAPE    M      FSKIP
INE GRAPH                          $BAR GRAPH PLOTTING FOR SUBROUT M      HSTPLT-II
 OR SUM POWER OF DEVIATIONS FROM BASE     $RAISE VECTOR TO POWER M      POWER
    $R.M.S. DEVIATION FROM GIVEN BASE OR FROM TRUE AVERAGE       M      RMSDEV
     $OPTIONAL ONLINE MONITOR OF BCD TAPE WRITING               M      ONLINE
                          $WRITE BINARY DATA ON TAPE            M      WRTDAT
      $READ EVERY N-TH WORD FROM BINARY TAPE                    N      PACDAT
                $CHANGE ALL SIGN BITS OF A VECTOR               M      CHSIGN
                     $MOVE DATA BLOCK                           M      MVBLOK
N LEVEL$SCAN VECTOR FOR POSSIBLE BLOCK OF VALUES ALL ABOVE GIVE F      NXALRM
UMMATION OF VECTOR OVER ABUTTING BLOCKS OF CONSTANT LENGTH   $S M      BLKSUM
AND OPERATE SUBROUTINES BY PROXY CALL STATEMENTS       $LOCATE  M      LOCATE
        $GET HOLLERITH DATA FROM CALLING SEQUENCE               F      GETHOL
 $ENABLE FORTRAN VARIABLE LENGTH CALLING SEQUENCES              M      VARARG
GE                          $SPACE CARRIAGE N LINES OR RESTORE PA F      CARIGE
RIC 2-DIMENSIONAL ARRAY  $ROTATE CENTRO-SYMMETRIC OR ANTISYMMET F      ROAR2
OR                                 $CHANGE ALL SIGN BITS OF A VECT M      CHSIGN
VOVE,REVERSE, CHANGE SPACING, OR CHANGE SIGN OF A VECTOR      $ M      MOVREV
 OF A VECTOR      $MOVE,REVERSE, CHANGE SPACING, OR CHANGE SIGN M      MOVREV
             $GENERATE HOLLERITH CHARACTERS                     M      GNHOL2
MAKING ON-LINE REQUEST IF NOT   $CHECK IF INTERVAL TIMER IS ON  F      CLKON
 FALL WITHIN GIVEN LIMITS       $CHECK THAT VARIABLES FROM LIST M      LIMITS
EASING OR DECREASING BEHAVIOR   $CHECK VECTOR FOR MOMOTONE INCR M      MONOCK
ILITY CASE                $COMPUTE CHI-SQUARE FOR CONSTANT PROBAB F      CHISQR
VALUE         $PROBABILITY THAT A CHI-SQUARED VARIATE EXCEEDS A  F      KIINT1
Y A THIRD ONE BEING ZERO        $CHOOSE BETWEEN TWO VARIABLES B M      WHICH
LE VECTOR TO INTEGERS FOR SCOPE, CLIPPING EXCESSIVE VALUES $SCA M      SCPSCL
G IN SECONDS USING 7090 INTERVAL CLOCK     $FOR REAL TIME TIMIN M 7090CLOCK1
$EVALUATE A POLYNOMIAL WITH REAL COEFFICIENTS FOR REAL ARGUMENT F      POLYEV
BOUT ITS MIDPOINT                  $COLLAPSE ODD-LENGTHED VECTOR A M      KOLAPS
 SMALLER RANGE                     $COLLAPSE ONE-SIDED VECTOR INTO M      COLAPS
TERAL FORMATS              $OUTPUT COLUMN VECTORS BY NORMAL OR LI M      CVSOUT
 INTERPOLATION     $FIND A MATRIX COLUMN WITH ARBITRARY INDEX BY M      ARBCOL
T INTEGERS         $LABEL PRINTER COLUMNS WITH INCREASING 3-DIGI F      COLABL
$LOCATE ARGUMENT WITH RESPECT TO COMMON                         F      IXCARG
NT OF MEMORY USAGE - PROGRAM AND COMMON           $OFF-LINE PRI F      MEMUSE
                  $FIND LENGTH OF COMMON STORAGE                M      XLCOMN
DS WHERE -0 IS LESS THAN +0     $COMPARE ARITHMETICALLY TWO WOR M      CMPRA
A SET OF VARIABLES FOR EQUALITY $COMPARE PAIRS OF VARIABLES OR  M      CMPARP
TORS FOR IDENTITY          $FAST COMPARE TWO ARBITRARY MODE VEC M      CMPARV
                           $COMPLEX POLYNOMIAL EVALUATION       F      IPLYEV
LYNOMIAL SYNTHESIS FROM REAL AND COMPLEX ROOTS          $PO F      POLYSN
AL SYNTHESIZED FROM ITS REAL AND COMPLEX ROOTS     $POLYNOMI F      PLYSYN
                     $TEST THE CONDITION OF ANY SENSE SWITCH    M      SWITCH
  $DIVIDE A FLOATING VECTOR BY A CONSTANT                       M      DIVIDE
 VARIABLES BY A SINGLE FLTG. PT. CONSTANT  $MULTIPLY ANY NO. OF F      MULK  -II
F VECTOR OVER ABUTTING BLOCKS OF CONSTANT LENGTH   $SUMMATION O M      BLKSUM
```

$MODIFY A SET OF VARIABLES BY A CONSTANT OR BY CONSTANTS	M	ADDK
$COMPUTE CHI-SQUARE FOR CONSTANT PROBABILITY CASE	F	CHISQR
OR FLTG VECTOR $ADD A CONSTANT TO ELEMENTS OF A FXD	M	BOOST
$DIVIDE A FXD VECTOR BY A CONSTANT	M	XDVIDE
OF TWO VECTORS WITH DIVISION BY CONSTANT $DOT PRODUCT	M	VDOTV
ING SUMMATION WITH DIVISION BY A CONSTANT $MOV	M	MVNSUM
IPLY VECTOR BY FLOATING OR FIXED CONSTANT $MULT	M	MULPLY
XED OR FLOATING VECTOR THROUGH A CONSTANT $REFLECT A FI	M	REFLEC
OF VECTOR FROM ANOTHER OR FROM A CONSTANT $SUM DIFFERENCE	M	SUMDFR
LL ELEMENTS OF VECTOR EQUAL TO A CONSTANT (ANY MODE) $SET A	M	SETKV
TG VECTOR FROM ANOTHER OR FROM A CONSTANT$SUM SQUARE DIF. OF FL	M	SQRDFR
D. VECTOR FROM ANOTHER OR FROM A CONSTANT$SUM SQUARE DIF. OR FX	M	XSQDFR
OF VARIABLES BY A CONSTANT OR BY CONSTANTS $MODIFY A SET	M	ADDK
M PROBABILITY DENSIT$MEAN SQUARE CONTINGENCY AND DEPENDENCY FRO	F	MSCON1
R IN DECIBELS $CONTOUR A MATRIX ON THE PRINTE	F	CNTRDB
ROW OF DATA $FIND CONTOUR LEVELS FOR PLOTTING A	F	CNTROW
F-LINE PRINTER $CONTOUR OF MATRIX SUBSET ON OF	F	CONTUR
E INTEGERS OR CONVERSELY $SCALE, CONVERT FLTG. VECTOR TO MACHIN	M	FXDATA
TO MLI VECTOR $FAST CONVERT FORTRAN INTEGER VECTOR	M	ITOMLI
ER TO EQUIVALENT HOLLERITH $CONVERT MACHINE LANGUAGE INTEG	M	MLI2A6
$COMPLETE CONVOLUTION OF TWO TRANSIENTS	M	CONVLV-II
$COMPLETE CONVOLUTION OF TWO TRANSIENTS	F	CONVLV
ACCURACY SERIES $FAST CONVOLUTIONS FOR LONG, LIMITED	F	QCNVLV
THER - VERSION 2 $FAST COPY FILE FROM ONE TAPE TO ANO	M	CPYFL2
RGUMENT $CORE LOCATION WITH INDEXABLE A	M	LOC
COSINE, SINE TRANSFORMS OF CROSS-CORRELATION FUNCTIONS $FAST	F	XSPECT
RA $MODIFY AUTO- OR CROSS-CORRELATIONS FOR DANIELL SPECT	M	ADANL
ACCURACY SERIES $FAST CROSS-CORRELATIONS FOR LONG, LIMITED	F	QXCORR
F FIXED POINT INTEGERS $FAST CORRELATIONS FOR LONG SERIES O	M	PROCOR
FROM 2 OR 4 EVEN-ODD PARTS $FAST COSINE AND/OR SINE TRANSFORMS	M	COSP
OF ODD-LENGTH SERIES $FAST COSINE AND/OR SINE TRANSFORMS	F	COSIS1
S, FIXED OR FLOATING $GENERATE COSINE OR SINE HALF-WAVE TABLE	M	COSTBL
ATION $AUTOSPECTRUM BY COSINE TRANSFORM OF AUTOCORREL	M	ASPEC2
AUTOCORRELATIONS $FAST COSINE TRANSFORMS OF ONE-SIDED	M	ASPECT
SS-CORRELATION FUNCTIONS $FAST COSINE, SINE TRANSFORMS OF CRO	F	XSPECT
NCTIONS FOR SEQUENTIAL SINES AND COSINES $FAST FU	M	SEQSAC
SERIES IN GIVEN RANGE$FREQUENCY COUNT OF NUMBER OF VALUES OF A	M	FRQCT2
WITH NEW RANGE AND INCREMENT $CREATE ONE VECTOR FROM ANOTHER	M	NURINC
SSES OF VARIABLES IN A LIST $CREATE VECTOR OF MACHINE ADDRE	M	XLOCV
$FAST COSINE, SINE TRANSFORMS OF CROSS-CORRELATION FUNCTIONS	F	XSPECT
SPECTRA $MODIFY AUTO- OR CROSS-CORRELATIONS FOR DANIELL	M	ADANL
IMITED ACCURACY SERIES $FAST CROSS-CORRELATIONS FOR LONG, L	F	QXCORR
VECTORS OF MATRICES $CROSSCORRELATION OF TRANSIENT	F	CRSVM
BEGINNING WITH ANY LAG $CROSSCORRELATION OF TRANSIENTS	F	CROST
BEGINNING WITH ZERO LAG $CROSSCORRELATION OF TRANSIENTS	F	CROSS
NAL SPATIAL ARRAYS $SPATIAL CROSSCORRELATION OF 2-DIMENSIO	F	SPCOR2
ENTS $QUICK CROSSCORRELATION OF MLI TRANSI	F	QXCOR1
SUBROUTINE GRAPH $CUBIC CURVE SCOPE PLOTTING FOR	M	HSTPLT-III
SUBROUTINE GRAPH $CUBIC CURVE SCOPE PLOTTING FOR	M	HSTPLT-III
NTS $FAST EVALUATE CUBIC FOR EVENLY SPACED ARGUME	M	FASCUB

```
PEED EXPANSION OF A VECTOR UNDER CUBIC INTERPOLATION      $HI-S M     EXPAND
ALLY SPACED POINTS          $FIND CUBIC WHICH EXACTLY FITS 4 EQU M     CUFIT1
UTINE GRAPH                 $CUBIC CURVE SCOPE PLOTTING FOR SUBRO M     HSTPLT-III
UTINE GRAPH                 $CUBIC CURVE SCOPE PLOTTING FOR SUBRO M     HSTPLT-III
 AUTO- OR CROSS-CORRELATIONS FOR DANIELL SPECTRA         $MODIFY M     ADANL
                            $MOVE DATA BLOCK                    M     MVBLOK
                            $READ DATA IN GENERALIZED FORMAT    F     RDATA
                   $WRITE BINARY DATA ON TAPE                   M     WRTDAT
CALE AND FIX DATA VECTOR, PACK N DATA POINTS PER REGISTER    $S M     PAKN
OR                        $REREAD DATA RECORD AND END FILE MONIT M     REREAD
              $FAST AND CONVIENT DATA STORAGE ON TAPE           F     OUDATA
    $UNPACK AND RESCALE A PACKED DATA VECTOR                    M     UNPAKN
S PER REGISTER    $SCALE AND FIX DATA VECTOR, PACK N DATA POINT M     PAKN
NTOUR A MATRIX ON THE PRINTER IN DECIBELS                 $CO F     CNTRDB
 AND REPOSITION TAPE TO FRONT OF DECK          $LIST DATA DECK F     DADECK
ONT OF DECK           $LIST DATA DECK AND REPOSITION TAPE TO FR F     DADECK
ECTOR FOR MOMOTONE INCREASING OR DECREASING BEHAVIOR   $CHECK V M     MONOCK
OF FUNCTION OR ITS MAGNITUDE    $DEFINITE TRAPEZOIDAL INTEGRAL  M     TINGL
ONS, FLOATING AND FIXED POINT   $DELTA FUNCTION AND STEP FUNCTI M     DELTA
 AND DEPENDENCY FROM PROBABILITY DENSIT$MEAN SQUARE CONTINGENCY F     MSCON1
IVEN LAG     $SECOON PROBABILITY DENSITY OF INTEGER SERIES AT G F     PROB2
NSIT$MEAN SQUARE CONTINGENCY AND DEPENDENCY FROM PROBABILITY DE F     MSCON1
ERENCING                        $DERIVATIVE OF A VECTOR OF DIFF M     DERIVA
ON OF SIMULTANEOUS EQUATIONS AND DETERMINANT EVALUATION $SOLUTI M     SIMEQ
ROM TRUE AVERAGE          $R.M.S. DEVIATION FROM GIVEN BASE OR F M     RMSDEV
 VECTOR TO POWER OR SUM POWER OF DEVIATIONS FROM BASE    $RAISE M     POWER
ECTOR ELEMENTS IN PAIRS         $DIFFERENCE FIXED OR FLOATING V M     DIFPRS
 IF SAME INCLUDING SIGN $SIGN OF DIFFERENCE OF 2 VARIABLES OR 0 M     XACTEQ
HER OR FROM A CONSTANT      $SUM DIFFERENCE OF VECTOR FROM ANOT M     SUMDFR
ER OR FROM A CONSTANT$SUM SQUARE DIF. OF FLTG VECTOR FROM ANOTH M     SQRDFR
ER OR FROM A CONSTANT$SUM SQUARE DIF. OR FXD. VECTOR FROM ANOTH M     XSQDFR
      $DERIVATIVE OF A VECTOR OF DIFFERENCING               M     DERIVA
$INVERSION OF DIFFERENTIATION BY DIFFERENCING               M     IDERIV
G                  $INVERSION OF DIFFERENTIATION BY DIFFERENCIN M     IDERIV
RINTER COLUMNS WITH INCREASING 3-DIGIT INTEGERS       $LABEL P F     COLABL
NE FOR RAND CORP. MILLION RANDOM DIGITS FROM TAPE $ACCESS ROUTI F     GETRD1
QUARES PREDICTOR BY RECURSION, 1-DIMENSION  $REALIZABLE LEAST S F     RLSPR
     $DISPLACED DOT PRODUCT OF 2-DIMENSIONAL ARRAYS         F     DOTP
N                          $TWO-DIMENSIONAL FILTER BY RECURSIO F     FIRE2
TRO-SYMMETRIC OR ANTISYMMETRIC 2-DIMENSIONAL ARRAY  $ROTATE CEN F     ROAR2
  $SPATIAL CROSSCORRELATION OF 2-DIMENSIONAL SPATIAL ARRAYS     F     SPCOR2
                   $FAST TWO-DIMENSIONAL SPATIAL SPECTRUM       F     PLANSP
QUARES PREDICTOR BY RECURSION, 2-DIMENSIONS $REALIZABLE LEAST S F     RLSPR2
T GENERATOR FOR SCOPE SUBROUTINE DISPLA  $VARIABLE ORIGIN FORMA M     DSPFMT
ENSIONAL ARRAYS                 $DISPLACED DOT PRODUCT OF 2-DIM F     DOTP
VECTOR                $FREQUENCY DISTRIBUTION OF A FIXED POINT  F     FRQCT1
 EQUALLY LIKELY SECTIONS $NORMAL DISTRIBUTION AND DIVISION INTO M     NOINT1
$REGION TO MAXIMIZE RATIO OF TWO DISTRIBUTION FUNCTIONS         F     MXRARE
MENTS      $GENERATE PROBABILITY DISTRIBUTION WITH SPECIFIED MO F     PRBFIT
CONSTANT                        $DIVIDE A FLOATING VECTOR BY A  M     DIVIDE
```

```
 PROBABLE RANGES                  $DIVIDE THE X AXIS INTO EQUALLY F     GRUP2
ANT                               $DIVIDE A FXD VECTOR BY A CONST M     XDVIDE
BY THOSE OF ANOTHER               $DIVIDE ELEMENTS OF ONE VECTOR  M     VDVBYV
ECTORS WITH OR WITHOUT ROUNDING $DIVIDE ELEMENTS OF TWO FIXED V M     XVDVBV
DING TO FORTRAN-II INTEGE$FXD PT DIVIDE WITH TRUNCATION OR ROUN M     XDIV
          $MOVING SUMMATION WITH DIVISION BY A CONSTANT          M     MVNSUM
$DOT PRODUCT OF TWO VECTORS WITH DIVISION BY CONSTANT            M     VDOTV
ECTIONS $NORMAL DISTRIBUTION AND DIVISION INTO EQUALLY LIKELY S M     NOINT1
                  $PERFORM LONG DIVISION OF TWO POLYNOMIALS    F     POLYDV
RRAYS                $DISPLACED DOT PRODUCT OF 2-DIMENSIONAL A F     DOTP
                        $FAST DOT PRODUCT OF TWO VECTORS      M     FDOT
ICES     $DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATR F     MDOT3
ICES     $DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATR F     MDOT
ODUCT OF VECTORS OF MATRICES     $DOT PRODUCT OR REVERSED DOT PR F     MDOT3
ODUCT OF VECTORS OF MATRICES     $DOT PRODUCT OR REVERSED DOT PR F     MDOT
REMENTS                       $VECTOR DOT PRODUCT WITH ARBITRARY INC M     DOTJ
H DIVISION BY CONSTANT           $DOT PRODUCT OF TWO VECTORS WIT M     VDOTV
R (FIXED OR FLOATING)       $FAST DOUBLING OR HALVING OF A VECTO M     DUBLX
GIVEN VALUE$FAST SCAN VECTOR FOR ELEMENT EQUAL OR GREATER THAN   M     FASCN1
      $EXTREMAL VALUES OF MATRIX ELEMENTS                        M     MAXSNM
FERENCE FIXED OR FLOATING VECTOR ELEMENTS IN PAIRS          $DIF M     DIFPRS
OR             $ADD A CONSTANT TO ELEMENTS OF A FXD OR FLTG VECT M     BOOST
R RIGHT             $SHIFT VECTOR ELEMENTS ARITHMETICALLY LEFT O M     SHFTR1
KE INDEX (BY INCREASING SIZE) OF ELEMENTS IN A VECTOR  $FAST MA M     SIZEUP
HT                  $SHIFT VECTOR ELEMENTS LOGICALLY LEFT OR RIG M     SHFTR2
OR                $SUM THE SQUARED ELEMENTS OF A FLTG OR FXD VECT M     SQRSUM
 INTEGER VECTOR     $FAST SQUARE ELEMENTS OF A MACHINE LANGUAGE M     SQRMLI
VECTOR                      $SUM ELEMENTS OF FLOATING OR FIXED   M     SUM
                         $SQUARE ELEMENTS OF FXD OR FLTG VECTOR M     SQUARE
E OF ANOTHER                $DIVIDE ELEMENTS OF ONE VECTOR BY THOS M     VDVBYV
WITH OR WITHOUT ROUNDING $DIVIDE ELEMENTS OF TWO FIXED VECTORS   M     XVDVBV
OR FLOATING               $MULTIPLY ELEMENTS OF TWO VECTORS FIXED M     VTIMSV
CONSTANT (ANY MODE)      $SET ALL ELEMENTS OF VECTOR EQUAL TO A  M     SETKV
 CALLING SEQUENCES               $ENABLE FORTRAN VARIABLE LENGTH M     VARARG
RTRAN                            $ENABLE MIXED EXPRESSIONS IN FO M     SAME
 AVERAGING, MOVING LEFT OR RIGHT END                  $TRIANGULAR M     TAMVL
         $REREAD DATA RECORD AND END FILE MONITOR                 M     REREAD
E   $TEST IF NEXT TAPE RECORD IS END OF FILE AND REPOSITION TAP M     ZEFBCD
LUE$FAST SCAN VECTOR FOR ELEMENT EQUAL OR GREATER THAN GIVEN VA M     FASCN1
       $PRINTER PLOT OF A SET OF EQUAL LENGTH VECTORS           F     PLTVS1
      $SET ALL ELEMENTS OF VECTOR EQUAL TO A CONSTANT (ANY MODE) M     SETKV
         $ SET FXD OR FLTG VECTOR EQUAL TO A LINEAR SEGMENT     M     SETLIN
R FLTG)$SET ANY NO. OF VARIABLES EQUAL TO A SINGLE VALUE (FXD O F     SETK  -II
OR FLTG) $SET ANY NO. OF VECTORS EQUAL TO SEPARATE VALUES (FXD  M     SETKVS
OR FLTG$SET ANY NO. OF VARIABLES EQUAL TO SEPARATE VALUES (FXD  F     SETKS -II
IABLES OR A SET OF VARIABLES FOR EQUALITY $COMPARE PAIRS OF VAR M     CMPARP
         $DIVIDE THE X AXIS INTO EQUALLY PROBABLE RANGES         F     GRUP2
$FIND CUBIC WHICH EXACTLY FITS 4 F UALLY SPACED POINTS           M     CUFIT1
L DISTRIBUTION AND DIVISION INTO EQUALLY LIKELY SECTIONS $NORMA M     NOINT1
D QUADRATIC WHICH EXACTLY FITS 3 EQUALLY SPACED POINTS      $FIN M     QUFIT1
```

```
UATION $SOLUTION OF SIMULTANEOUS EQUATIONS AND DETERMINANT EVAL M      SIMEQ
VERT MACHINE LANGUAGE INTEGER TO EQUIVALENT HOLLERITH      $CON M      MLI2A6
   $WIENER-LEVINSON LEAST SQUARE ERROR FILTER OR PREDICTOR      F      WLLSFP
ED ARGUMENTS                $FAST EVALUATE CUBIC FOR EVENLY SPAC M      FASCUB
L COEFFICIENTS FOR REAL ARGUMENT$EVALUATE A POLYNOMIAL WITH REA F      POLYEV
          $COMPLEX POLYNOMIAL EVALUATION                        F      IPLYEV
ANEOUS EQUATIONS AND DETERMINANT EVALUATION $SOLUTION OF SIMULT M      SIMEQ
 IN GROUPS OF FIVE AS POKER HAND$EVALUATION OF INTEGER SEQUENCE F      POKCT1
)       $SPLIT A VECTOR INTO ITS EVEN AND ODD PARTS (OR INVERSE M      SPLIT
M A VECTOR BY SIFTING ANOTHER AT EVEN INCREMENTS          $FOR M      SIFT
NE WHETHER FORTRAN-II INTEGER IS EVEN OR ODD          $DETERMI M      XOOZE
D/OR SINE TRANSFORMS FROM 2 OR 4 EVEN-ODD PARTS $FAST COSINE AN M      COSP
        $FAST EVALUATE CUBIC FOR EVENLY SPACED ARGUMENTS        M      FASCUB
NTERPOLATION OPERATOR FOR 1 TO 4 EVENLY SPACED DATA VALUES   $I M      INTOPR
 TO INTEGERS FOR SCOPE, CLIPPING EXCESSIVE VALUES $SCALE VECTOR M      SCPSCL
                    $EXCHANGE ANY TWO VECTORS                 M      EXCHVS
             $SUBROUTINE GRAPH EXPANDED OVER VERTICAL FRAMES  F      GRAPHX
BIC INTERPOLATION      $HI-SPEED EXPANSION OF / VECTOR UNDER CU M      EXPAND
                $ENABLE MIXED EXPRESSIONS IN FORTRAN          M      SAME
        $FIND SIGNED OR UNSIGNED EXTREMAL VALUES OF A VECTOR  M      MAXSN
ENTS                        $EXTREMAL VALUES OF MATRIX ELEM M      MAXSNM
FINITE MATRIX               $FACTOR A SYMMETRIC POSITIVE DE F      MFACT
MINIMUM PHASE WAVELET       $FACTOR POWER SPECTRUM TO FIND  M      FACTOR
R                           $FAST ABSOLUTE VALUE OF A VECTO M      ABSVL
OF DATA FROM A SPECIAL TAPE $FAST A' CONVENIENT RETRIEVAL   F      INDATA
EGMENT ON SCOPE             $FAST ARBITRARY STRAIGHT LINE S M 7090LINE
EGMENT ON SCOPE             $FAST ARBITRARY STRAIGHT LINE S M  709LINE
E VECTORS FOR IDENTITY      $FAST COMPARE TWO ARBITRARY MOD M      CMPARV
ECTOR TO MLI VECTOR         $FAST CONVERT FORTRAN INTEGER V M      ITOMLI
O ANOTHER - VERSION 2       $FAST COPY FILE FROM ONE TAPE T M      CPYFL2
ORMS FROM 2 OR 4 EVEN-ODD PARTS $FAST COSINE AND/OR SINE TRANSF M      COSP
ORMS OF ODD-LENGTH SERIES   $FAST COSINE AND/OR SINE TRANSF F      COSIS1
SIDED AUTOCORRELATIONS      $FAST COSINE TRANSFORMS OF ONE- M      ASPECT
S                           $FAST DOT PRODUCT OF TWO VECTOR M      FDOT
VECTOR (FIXED OR FLOATING)  $FAST DOUBLING OR HALVING OF A  M      DUBLX
 SPACED ARGUMENTS           $FAST EVALUATE CUBIC FOR EVENLY M      FASCUB
                       $PLOT FAST HORIZONTAL LINE ON SCOPE  M 7090LINEH
                       $PLOT FAST HORIZONTAL LINE ON SCOPE  M  709LINEH
RS ,AS PRODUCED BY SPLIT.   $FAST REVERSAL OF SPECIAL VECTO M      CHPRTS
QUAL OR GREATER THAN GIVEN VALUE$FAST SCAN VECTOR FOR ELEMENT E M      FASCN1
 INDICES                    $FAST TRACK THROUGH A VECTOR OF M      FASTRK
                       $PLOT FAST VERTICAL LINE ON SCOPE    M 7090LINEV
                       $PLOT FAST VERTICAL LINE ON SCOPE    M  709LINEV
 ON TAPE                    $FAST AND CONVIENT DATA STORAGE F      OUDATA
, LIMITED ACCURACY SERIES   $FAST AUTOCORRELATIONS FOR LONG F      QACORR
MITED ACCURACY SERIES       $FAST CONVOLUTIONS FOR LONG, LI F      QCNVLV
IES OF FIXED POINT INTEGERS $FAST CORRELATIONS FOR LONG SER M      PROCOR
F CROSS-CORRELATION FUNCTIONS $FAST COSINE, SINE TRANSFORMS O F      XSPECT
NG, LIMITED ACCURACY SERIES $FAST CROSS-CORRELATIONS FOR LO F      QXCORR
SIENT WITH ARBITRARY TIME ORIGIN$FAST FOURIER TRANSFORM OF TRAN F      QFURRY
```

```
SINES AND COSINES                    $FAST FUNCTIONS FOR SEQUENTIAL  M      SEQSAC
 SIZE) OF ELEMENTS IN A VECTOR       $FAST MAKE INDEX (BY INCREASING M      SIZEUP
ED POINT VECTOR                      $FAST MOVING SUMMATION OF A FIX M      MUVADD
A VECTOR                             $FAST REVERSE STORAGE ORDER OF  M      REVERS
                                     $FAST SET VECTOR TO ZERO        M      STZ
INE LANGUAGE INTEGER VECTOR          $FAST SQUARE ELEMENTS OF A MACH M      SQRMLI
PECTRUM                              $FAST TWO-DIMENSIONAL SPATIAL S F      PLANSP
            $GENERATE HOLLERITH FIELD                                M      GENHOL
- VERSION 2           $FAST COPY FILE FROM ONE TAPE TO ANOTHER       M      CPYFL2
ST IF NEXT TAPE RECORD IS END OF FILE AND REPOSITION TAPE   $TE M      ZEFBCD
     $REREAD DATA RECORD AND END FILE MONITOR                    M      REREAD
  $SKIP FORWARD OR BACKWARD OVER FILES ON TAPE                   M      FSKIP
                       $ADVANCE FILM FRAME ON SCOPE              M 7090FRAME
                       $ADVANCE FILM FRAME ON SCOPE              M  709FRAME
                  $MULTI-INPUT FILTER BY LEAST SQUARES           F      MIFLS
               $TWO-DIMENSIONAL FILTER BY RECURSION              F      FIRE2
SPONSE     $GENERATE SYMMETRICAL FILTER WITH GIVEN AMPLITUDE RE F      GNFLT1
ENER-LEVINSON LEAST SQUARE ERROR FILTER OR PREDICTOR        $WI F      WLLSFP
       $FIND CUBIC WHICH EXACTLY FITS 4 EQUALLY SPACED POINTS  M      CUFIT1
   $FIND QUADRATIC WHICH EXACTLY FITS 3 EQUALLY SPACED POINTS  M      QUFIT1
WITHOUT ROUNDING                     $FIX A FLOATING VECTOR WITH OR  M      FIXV
OINTS PER REGISTER    $SCALE AND FIX DATA VECTOR, PACK N DATA P M      PAKN
DOUBLING OR HALVING OF A VECTOR (FIXED OR FLOATING)      $FAST  M      DUBLX
COSINE OR SINE HALF-WAVE TABLES, FIXED OR FLOATING   $GENERATE  M      COSTBL
NTS IN PAIRS          $DIFFERENCE FIXED OR FLOATING VECTOR ELEME M      DIFPRS
AND STEP FUNCTIONS, FLOATING AND FIXED POINT   $DELTA FUNCTION  M      DELTA
TIPLY AN MLI VECTOR BY A FORTRAN FIXED POINT INTEGER       $MUL M      MLISCL
    $FREQUENCY DISTRIBUTION OF A FIXED POINT VECTOR             F      FRQCT1
RATED SUMMATION OF A FLOATING OF FIXED VECTOR            $INTEG M      INTSUM
           $SET LINEAR VECTORS, FIXED AND/OR FLOAT+NG           M      SETLNS
 $MULTIPLY VECTOR BY FLOATING OR FIXED CONSTANT                 M      MULPLY
MULTIPLY ELEMENTS OF TWO VECTORS FIXED OR FLOATING             $ M      VTIMSV
GH A CONSTAN.          $REFLECT A FIXED OR FLOATING VECTOR THROU M      REFLEC
 CORRELATIONS FOR LONG SERIES OF FIXED POINT INTEGERS     $FAST M      PROCOR
     $FAST MOVING SUMMATION OF A FIXED POINT VECTOR             M      MUVADD
               $FIND AVERAGE OF FIXED PT VECTOR                 M      XAVRGE
        $REMOVE THE MEAN FROM A FIXED VECTOR                    M      XREMAV
    $SUM FLEMENTS OF FLOATING OR FIXED VECTOR                   M      SUM
              $SQUARE ROOT OF A FIXED VECTOR WITH ROUNDING      M      XSQRUT
$ADD OR SUBTRACT TWO FLOATING OR FIXED VECTORS                  M      VPLUSV
ROUNDING $DIVIDE ELEMENTS OF TWO FIXED VECTORS WITH OR WITHOUT  M      XVDVBV
                                     $FLOAT A VECTOR            M      FLOATV
EGER                                 $FLOAT ANY MACHINE LANGUACE INT M      FLOATM
OR HALVING OF A VECTOR (FIXED OR FLOATING)      $FAST DOUBLING  M      DUBLX
 SINE HALF-WAVE TABLES, FIXED OR FLOATING   $GENERATE COSINE OR M      COSTBL
LTA FUNCTION AND STEP FUNCTIONS, FLOATING AND FIXED POINT   $DE M      DELTA
      $INTEGRATED SUMMATION OF A FLOATING OF FIXED VECTOR       M      INTSUM
               $FIND AVERAGE OF FLOATING VECTOR                 M      AVRAGE
                    $DIVIDE A FLOATING VECTOR BY A CONSTANT     M      DIVIDE
IRS         $DIFFERENCE FIXED OR FLOATING VECTOR ELEMENTS IN PA M      DIFPRS
```

```
T ROUNDING                    $FIX A FLOATING VECTOR WITH OR WITHOU M      FIXV
SET LINEAR VECTORS, FIXED AND/OR FLOAT+NG                        $ M      SETLNS
FLEMENTS OF TWO VECTORS FIXED OR FLOATING                $MULTIPLY  M      VTIMSV
              $MULTIPLY VECTOR BY FLOATING OR FIXED CONSTANT       M      MULPLY
                 $SUM ELEMENTS OF FLOATING OR FIXED VECTOR         M      SUM
             $ADD OR SUBTRACT TWO FLOATING OR FIXED VECTORS        M      VPLUSV
 INTEGER       $TRUNCATE OR ROUND FLOATING PT. NUMBER TO MACHINE M      XFIXM
         $REMOVE THE MEAN FROM A FLOATING VECTOR                  M      REMAV
ROUND, ROUND UP, OR ROUND DOWN A FLOATING VECTOR                 $ M      RNDV
               $SQUARE ROOT OF A FLOATING VECTOR                  M      SQROOT
TANT          $REFLECT A FIXED OR FLOATING VECTOR THROUGH A CONS M      REFLEC
CONSTANT TO ELEMENTS OF A FXD OR FLTG VECTOR             $ADD A  M      BOOST
ANY NO. OF VARIABLES BY A SINGLE FLTG. PT. CONSTANT  $MULTIPLY  F      MULK  -II
RS OR CONVERSELY $SCALE, CONVERT FLTG. VECTOR TO MACHINE INTEGE M      FXDATA
 EQUAL TO A SINGLE VALUE (FXD OR FLTG)$SET ANY NO. OF VARIABLES F      SETK  -II
EQUAL TO SEPARATE VALUES (FXD OR FLTG) $SET ANY NO. OF VECTORS  M      SETKVS
  $SUM THE SQUARED ELEMENTS OF A FLTG OR FXD VECTOR              M      SQRSUM
      $SQUARE ELEMENTS OF FXD OR FLTG VECTOR                     M      SQUARE
SEGMENT              $ SET FXD OR FLTG VECTOR EQUAL TO A LINEAR  M      SETLIN
OM A CONSTANT$SUM SQUARE DIF. OF FLTG VECTOR FROM ANOTHER OR FR M      SQRDFR
EQUAL TO SEPARATE VALUES (FXD OR FLTG$SET ANY NO. OF VARIABLES  F      SETKS -II
 PT. NO. UP, DOWN, OR TO NEAREST FLTG. PT. INTEGER $ROUND FLTG. M      RND
NEAREST FLTG. PT. INTEGER $ROUND FLTG. PT. NO. UP, DOWN, OR TO  M      RND
  $ACCESS TO LITERAL OR ORDINARY FORMAT                          M      FNDFMT
            $MATRIX OUTPUT IN G FORMAT                          F      MOUT
PUT VARIABLES FIVE PER LINE IN G FORMAT                  $OUT M      CSOUT
ROUTINE DISPLA  $VARIABLE ORIGIN FORMAT GENERATOR FOR SCOPE SUB M      DSPFMT
TPUT TAPE WITH NORMAL OR LITERAL FORMAT VECTOR         $WRITE OU F      FMTOUT
OR OUTPUT WITH NORMAL OR LITERAL FORMAT            $OFFLINE VECT F      VECOUT
T VARIABLES BY NORMAL OR LITERAL FORMAT                  $OUTPU M      VRSOUT
       $READ DATA IN GENERALIZED FORMAT                          F      RDATA
R OUTPUT STATEMENT  $REPLACE THE FORMAT OF A SUCCEEDING INPUT O M      RPLFMT
AMED VECTOR BY NORMAL OR LITERAL FORMAT WITH SPACING  $OUTPUT N F      VOUT
UMN VECTORS BY NORMAL OR LITERAL FORMATS              $OUTPUT COL M      CVSOUT
MED VECTORS BY NORMAL OR LITERAL FORMATS WITH SPACING$OUTPUT NA M      VSOUT
    $MULTIPLY AN MLI VECTOR BY A FORTRAN FIXED POINT INTEGER     M      MLISCL
ERITH VECTOR             $PACK UP FORTRAN INTEGER VECTOR AS HOLL M      IVTOHV
VECTOR             $FAST CONVERT FORTRAN INTEGER VECTOR TO MLI  M      ITOMLI
 $SPREAD OUT HOLLERITH VECTOR AS FORTRAN INTEGERS               M      HVTOIV
    $ENABLE MIXED EXPRESSIONS IN FORTRAN                        M      SAME
E WITH TRUNCATION OR ROUNDING TO FORTRAN-II INTEGE$FXD PT DIVID M      XDIV
ODD           $DETERMINE WHETHER FORTRAN-II INTEGER IS EVEN OR  M      XOOZE
G SEQUENCES                $ENABLE FORTRAN VARIABLE LENGTH CALLIN M      VARARG
 WITH ARBITRARY TIME ORIGIN$FAST FOURIER TRANSFORM OF TRANSIENT F      QFURRY
RY TIME ORIGIN    $QUICK INVERSE FOURIER TRANSFORM WITH ARBITRA F      QIFURY
                   $ADVANCE FILM FRAME ON SCOPE                 M 7090FRAME
                   $ADVANCE FILM FRAME ON SCOPE                 M  709FRAME
TS                      $MULTIPLE FRAME SCOPE PLOTS OF VECTOR SE F      GRAPH
INE GRAPH EXPANDED OVER VERTICAL FRAMES                $SUBROUT F      GRAPHX
ALUES OF A SERIES IN GIVEN RANGE$FREQUENCY COUNT OF NUMBER OF V M      FRQCT2
```

```
XED POINT VECTOR                    $FREQUENCY DISTRIBUTION OF A FI F      FRQCT1
                      $ARCTANGENT FUNCTION                          M      ARCTAN
RITH LEFT ADJUST OR RIGHT ADJUST FUNCTION                $HOLLE M      HLADJ
                    $LOGICAL SHIFT FUNCTION                         M      LSHFT
LOATING AND FIXED POINT   $DELTA FUNCTION AND STEP FUNCTIONS, F M      DELTA
ON      $INVERSION OF A MONOTONE FUNCTION BY LINEAR INTERPOLATI M      IFNCTN
DEFINITE TRAPEZOIDAL INTEGRAL OF FUNCTION OR ITS MAGNITUDE    $ M      TINGL
POINT   $DELTA FUNCTION AND STEP FUNCTIONS, FLOATING AND FIXED  M      DELTA
 TRANSFORMS OF CROSS-CORRELATION FUNCTIONS   $FAST COSINE, SINE F      XSPECT
XIMIZE RATIO OF TWO DISTRIBUTION FUNCTIONS        $REGION TO MA F      MXRARE
 AND COSINES                $FAST FUNCTIONS FOR SEQUENTIAL SINES M      SEQSAC
$ADD A CONSTANT TO ELEMENTS OF A FXD OR FLTG VECTOR                M      BOOST
ECTORS EQUAL TO SEPARATE VALUES (FXD OR FLTG) $SET ANY NO. OF V M      SETKVS
RIABLES EQUAL TO A SINGLE VALUE (FXD OR FLTG)$SET ANY NO. OF VA F      SETK  -II
           $SQUARE ELEMENTS OF FXD OR FLTG VECTOR                  M      SQUARE
LINEAR SEGMENT             $ SET FXD OR FLTG VECTOR EQUAL TO A  M      SETLIN
IABLES EQUAL TO SEPARATE VALUES (FXD OR FLTG$SET ANY NO. OF VAR F      SETKS -II
OR ROUNDING TO FORTRAN-II INTEGE$FXD PT DIVIDE WITH TRUNCATION  M      XDIV
HE SQUARED ELEMENTS OF A FLTG OR FXD VECTOR              $SUM T M      SQRSUM
                    $DIVIDE A FXD VECTOR BY A CONSTANT          M      XDVIDE
OM A CONSTANT$SUM SQUARE DIF. OR FXD. VECTOR FROM ANOTHER OR FR M      XSQDFR
             $MATRIX OUTPUT IN G FORMAT                         F      MOUT
UTPUT VARIABLES FIVE PER LINE IN G FORMAT                 $O M      CSOUT
AVE TABLES, FIXED OR FLOATING   $GENERATE COSINE OR SINE HALF-W M      COSTBL
                                $GENERATE HOLLERITH CHARACTERS  M      GNHOL2
                                $GENERATE HOLLERITH FIELD       M      GENHOL
TH GIVEN AMPLITUDE RESPONSE     $GENERATE SYMMETRICAL FILTER WI F      GNFLT1
ION WITH SPECIFIED MOMENTS      $GENERATE PROBABILITY DISTRIBUT F      PRBFIT
 DISPLA  $VARIABLE ORIGIN FORMAT GENERATOR FOR SCOPE SUBROUTINE M      DSPFMT
AR GRAPH PLOTTING FOR SUBROUTINE GRAPH                    $B M      HSTPLT-II
VE SCOPE PLOTTING FOR SUBROUTINE GRAPH              $CUBIC CUR M      HSTPLT-III
VE SCOPE PLOTTING FOR SUBROUTINE GRAPH              $CUBIC CUR M      HSTPLT-III
ISTOGRAM PLOTTING FOR SUBROUTINE GRAPH                    $H M      HSTPLT
RAMES                 $SUBROUTINE GRAPH EXPANDED OVER VERTICAL F F      GRAPHX
GRAPH                       $BAR GRAPH PLOTTING FOR SUBROUTINE  M      HSTPLT-II
VALUATION OF INTEGER SEQUENCE IN GROUPS OF FIVE AS POKER HAND$E F      POKCT1
ATING   $GENERATE COSINE OR SINE HALF-WAVE TABLES, FIXED OR FLO M      COSTBL
FLOATING)      $FAST DOUBLING OR HALVING OF A VECTOR (FIXED OR  M      DUBLX
UENCE IN GROUPS OF FIVE AS POKER HAND$EVALUATION OF INTEGER SEQ F      POKCT1
 UNDER CUBIC INTERPOLATION      $HI-SPEED EXPANSION OF A VECTOR M      EXPAND
                                $HIGH SPEED 24 POINT SPECTRUM   F      FT24  -II
                                $HIGH SPEED 24 POINT SPECTRUM   M      FT24
INE GRAPH                       $HISTOGRAM PLOTTING FOR SUBROUT M      HSTPLT
E LANGUAGE INTEGER TO EQUIVALENT HOLLERITH      $CONVERT MACHIN M      ML12A6
                      $INTERPRET HOLLERITH                      M      INTHOL
                       $GENERATE HOLLERITH CHARACTERS           M      GNHOL2
QUENCE                      $GET HOLLERITH DATA FROM CALLING SE F      GETHOL
                       $GENERATE HOLLERITH FIELD                M      GENHOL
 ADJUST FUNCTION                $HOLLERITH LEFT ADJUST OR RIGHT M      HLADJ
                          $WRITE HOLLERITH TEXT ON SCOPE        M 7090DISPLA
```

$WRITE	HOLLERITH TEXT ON SCOPE	M	709DISPLA
ACK UP FORTRAN INTEGER VECTOR AS	HOLLERITH VECTOR $P	M	IVTOH
TEGERS $SPREAD OUT	HOLLERITH VECTOR AS FORTRAN IN	M	HVTOIV
F TWO ARBITRARY MODE VECTORS FOR	IDENTITY $FAST COMPAR	M	CMPARV
MPLITUDE AND PHASE FROM REAL AND	IMAGINARY, OR REVERSE $A	M	AMPHZ
$LABEL PRINTER COLUMNS WITH	INCREASING 3-DIGIT INTEGERS	F	COLABL
IOR $CHECK VECTOR FOR MONOTONE	INCREASING OR DECREASING BEHAV	M	MONOCK
N A VECTOR $FAST MAKE INDEX (BY	INCREASING SIZE) OF ELEMENTS I	M	SIZEUP
FROM ANOTHER WITH NEW RANGE AND	INCREMENT $CREATE ONE VECTOR	M	NURINC
TING $HYBRID SUBPROGRAMS FOR	INCREMENTING, TESTING, AND SET	M	INDEX
ECTOR DOT PRODUCT WITH ARBITRARY	INCREMENTS $V	M	DOTJ
ECTOR BY SIFTING ANOTHER AT EVEN	INCREMENTS $FORM A V	M	SIFT
IST AUXILIARY INFORMATION FOR AN	INDATA-OUDATA TYPE TAPE $L	F	LISTING
$INITIALIZED FOR ADDING TO AN	INDATA-OUDATA TAPE	F	SETINO
$TERMINATE AN	INDATA-OUDATA TAPE	F	TRMINO
IDAL RULE $INDEFINITE	INTEGRAL BY TRAPEZO	M	INTGRA
E A MATRIX COLUMN WITH ARBITRARY	INDEX BY INTERPOLATION $FIN	M	ARBCOL
ELEMENTS IN A VECTOR $FAST MAKE	INDEX (BY INCREASING SIZE) OF	M	SIZEUP
$CORE LOCATION WITH	INDEXABLE ARGUMENT	M	LOC
$ALLOWS VARIABLE DEPTH	INDEXING OF VECTORS	M	GETX
$FAST TRACK THROUGH A VECTOR OF	INDICES	M	FASTRK
NDATA-OUDATA TAPE $INITIALIZED	FOR ADDING TO AN I	F	SETINO
$MULTI-INPUT	FILTER BY LEAST SQUARES	F	MIFLS
ES $MULTI-INPUT	PREDICTOR BY LEAST SQUAR	F	MIPLS
$MULTI-INPUT	SIDEWARDS ITERATION	F	MISS
PLACE THE FORMAT OF A SUCCEEDING	INPUT OR OUTPUT STATEMENT $RE	M	RPLFMT
$FLOAT ANY MACHINE LANGUAGE	INTEGER	M	FLOATM
VECTOR BY A FORTRAN FIXED POINT	INTEGER $MULTIPLY AN MLI	M	MLISCL
AP A SEQUENCE OF NUMBERS INTO AN	INTEGER SERIES $M	M	MPSEQ1
H $CONVERT MACHINE LANGUAGE	INTEGER TO EQUIVALENT HOLLERIT	M	MLI2A6
CTOF $PACK UP FORTRAN	INTEGER VECTOR AS HOLLERITH VE	M	IVTOHV
$FAST CONVERT FORTRAN	INTEGER VECTOR TO MLI VECTOR	M	ITOMLI
ATION OR ROUNDING TO FORTRAN-II	INTEGER$FXD PT DIVIDE WITH TRUN	M	XDIV
E, DOWN, OR TO NEAREST FLTG. PT.	INTEGER $ROUND FLTG. PT. NO. U	M	RND
D FLOATING PT. NUMBER TO MACHINE	INTEGER $TRUNCATE OR ROUN	M	XFIXM
$DETERMINE WHETHER FORTRAN-II	INTEGER IS EVEN OR ODD	M	XOOZE
FIVE AS POKER HANDSEVALUATION OF	INTEGER SEQUENCE IN GROUPS OF	F	POKCT1
$SECOND PROBABILITY DENSITY OF	INTEGER SERIES AT GIVEN LAG	F	PROB2
F ELEMENTS OF A MACHINE LANGUAGE	INTEGER VECTOR $FAST SQUAR	M	SQRMLI
F OUTPUT TAPE A MACHINE LANGUAGE	INTEGER VECTOR $PRINT OR WRIT	F	PWMLIV
COLUMNS WITH INCREASING 3-DIGIT	INTEGERS $LABEL PRINTER	F	COLABL
OUT HOLLERITH VECTOR AS FORTRAN	INTEGERS $SPREAD	M	HVTOIV
NE $OUTPUT A MATRIX AS	INTEGERS DENSELY PACKED OFF-LI	F	MOUTAI
CONVERT FLTG. VECTOR TO MACHINE	INTEGERS OR CONVERSELY $SCALE,	M	FXDATA
S FOR LONG SERIES OF FIXED POINT	INTEGERS $FAST CORRELATION	M	PROCOR
XCESSIVE VALUES $SCALE VECTOR TO	INTEGERS FOR SCOPE, CLIPPING E	M	SCPSCL
$SHUFFLE A LIST OF	INTEGERS FROM 1 TO N	F	SHUFFL
$INVERSION OF TRAPEZOIDAL	INTEGRAL	M	IINTGR
$INDEFINITE	INTEGRAL BY TRAPEZOIDAL RULE	M	INTGRA
OIDAL INTEGRAL OR ABSOLUTE VALUE	INTEGRAL $MOVING TRAPEZ	M	MVNTIN

```
CALE OR SCALE VECTOR FOR SIMPSON INTEGRAL AND/OR INTEGRATE $UNS F      SMPSON
GNITUDE    $DEFINITE TRAPEZOIDAL INTEGRAL OF FUNCTION OR ITS MA M      TINGL
FGRAL        $MOVING TRAPEZOIDAL INTEGRAL OR ABSOLUTE VALUE INT M      MVNTIN
CTOR FOR SIMPSON INTEGRAL AND/OR INTEGRATE $UNSCALE OR SCALE VE F      SMPSON
TING OF FIXED VECTOR            $INTEGRATED SUMMATION OF A FLOA M      INTSUM
X COLUMN WITH ARBITRARY INDEX BY INTERPOLATION    $FIND A MATRI M      ARBCOL
XPANSION OF A VECTOR UNDER CUBIC INTERPOLATION      SHI-SPEED E M      EXPAND
OF A MONOTONE FUNCTION BY LINEAR INTERPOLATION      $INVERSION  M      IFNCTN
                        $LINEAR INTERPOLATION IN A TABLE        F      LINTR1
0 4 EVENLY SPACED DATA VALUES   $INTERPOLATION OPERATOR FOR 1 T M      INTOPR
                      $QUADRATIC INTERPOLATION IN A TABLE       F      QINTR1
                                $INTERPRET HOLLERITH            M      INTHOL
IME TIMING IN SECONDS USING 7090 INTERVAL CLOCK     $FOR REAL T M 7090CLOCK1
-LINE REQUEST IF NOT   $CHECK IF INTERVAL TIMER IS ON MAKING ON F      CLKON
                                $INVERSE OF A MATRIX            F      MATINV
 INTO ITS EVEN AND ODD PARTS (OR INVERSE)       $SPLIT A VECTOR M      SPLIT
 ARBITRARY TIME ORIGIN    $QUICK INVERSE FOURIER TRANSFORM WITH F      QIFURY
ON BY LINEAR INTERPOLATION      $INVERSION OF A MONOTONE FUNCTI M      IFNCTN
Y DIFFERENCING                  $INVERSION OF DIFFERENTIATION B M      IDERIV
RAL                             $INVERSION OF TRAPEZOIDAL INTEG M      IINTGR
LEAST SQUARES SHAPER BY SIDEWAYS ITERATION                    $ F      LSSS1
          $MULTI-INPUT SIDEWARDS ITERATION                    F      MISS
REASING 3-DIGIT INTEGERS        $LABEL PRINTER COLUMNS WITH INC F      COLABL
OF TRANSIENTS BEGINNING WITH ANY LAG         $CROSSCORRELATION  F      CROST
F TRANSIENTS BEGINNING WITH ZERO LAG       $CROSSCORRELATION 0 F      CROSS
              $FLOAT ANY MACHINE LANGUAGE INTEGER               M      FLOATM
 HOLLERITH      $CONVERT MACHINE LANGUAGE INTEGER TO EQUIVALENT M      MLI2A6
AST SQUARE ELEMENTS OF A MACHINE LANGUAGE INTEGER VECTOR     $F M      SQRMLI
T OR WRITE OUTPUT TAPE A MACHINE LANGUAGE INTEGER VECTOR  $PRIN F      PWMLIV
          $MULTI-INPUT FILTER BY LEAST SQUARES                F      MIFLS
        $MULTI-INPUT PREDICTOR BY LEAST SQUARES               F      MIPLS
                                $LEAST SQUARES LINE           F      LSLINE
YS ITERATION                    $LEAST SQUARES SHAPER BY SIDEWA F      LSSS1
REDICTOR        $WIENER-LEVINSON LEAST SQUARE ERROR FILTER OR P F      WLLSFP
URSION, 1-DIMENSION  $REALIZABLE LEAST SQUARES PREDICTOR BY REC F      RLSPR
URSION, 2-DIMENSIONS $REALIZABLE LEAST SQUARES PREDICTOR BY REC F      RLSPR2
ION                  $REALIZABLE LEAST SQUARES SHAPER BY RECURS F      RLSSR
NCTION                $HOLLERITH LEFT ADJUST OR RIGHT ADJUST FU M      HLADJ
$SHIFT VECTOR ELEMENTS LOGICALLY LEFT OR RIGHT                  M      SHFTR2
T VECTOR ELEMENTS ARITHMETICALLY LEFT OR RIGHT           $SHIF M      SHFTR1
   $TRIANGULAR AVERAGING, MOVING LEFT OR RIGHT END              M      TAMVL
 BLOCK OF VALUES ALL ABOVE GIVEN LEVEL$SCAN VECTOR FOR POSSIBLE F      NXALRM
ATA                $FIND CONTOUR LEVELS FOR PLOTTING A ROW OF D F      CNTROW
LTER OR PREDICTOR        $WIENER-LEVINSON LEAST SQUARE ERROR FI F      WLLSFP
$FAST AUTOCORRELATIONS FOR LONG, LIMITED ACCURACY SERIES        F      QACORR
    $FAST CONVOLUTIONS FOR LONG, LIMITED ACCURACY SERIES        F      QCNVLV
AST CROSS-CORRELATIONS FOR LONG, LIMITED ACCURACY SERIES     $F F      QXCORR
IND IF ARGUMENT FALLS INSIDE TWO LIMITING VALUES             $F M      XLIMIT
BLES FROM LIST FALL WITHIN GIVEN LIMITS       $CHECK THAT VARIA M      LIMITS
                $LEAST SQUARES LINE                           F      LSLI IE
```

```
X AS INTEGERS DENSELY PACKED OFF-LINE          $OUTPUT A MATRI F      MOUTAI
      $OUTPUT VARIABLES FIVE PER LINE IN G FORMAT                  M      CSOUT
           $PLOT FAST HORIZONTAL LINE ON SCOPE                     M 7090LINEH
           $PLOT FAST HORIZONTAL LINE ON SCOPE                     M  709LINEH
             $PLOT FAST VERTICAL LINE ON SCOPE                     M 7090LINEV
             $PLOT FAST VERTICAL LINE ON SCOPE                     M  709LINEV
ROGRAM AND COMMON          $OFF-LINE PRINT OF MEMORY USAGE - P F      MEMUSE
$CONTOUR OF MATRIX SUBSET ON OFF-LINE PRINTER                    F      CONTUR
F INTERVAL TIMER IS ON MAKING ON-LINE REQUEST IF NOT   $CHECK I F      CLKCN
        $FAST ARBITRARY STRAIGHT LINE SEGMENT ON SCOPE           M 7090LINE
        $FAST ARBITRARY STRAIGHT LINE SEGMENT ON SCOPE           M  709LINE
ERSION OF A MONOTONE FUNCTION BY LINEAR INTERPOLATION      $INV M      IFNCTN
E                              $LINEAR INTERPOLATION IN A TABL F      LINTR1
ET FXD OR FLTG VECTOR EQUAL TO A LINEAR SEGMENT             $ S M      SETLIN
LOAT+NG                        $SET LINEAR VECTORS, FIXED AND/OR F M      SETLNS
              $SPACE CARRIAGE N LINES OR RESTORE PAGE            F      CARIGE
 AN INDATA-OUDATA TYPE TAPE    $LIST AUXILIARY INFORMATION FOR F      LISTING
TAPE TO FRONT OF DECK          $LIST DATA DECK AND REPOSITION  F      DADECK
      $CHECK THAT VARIABLES FROM LIST FALL WITHIN GIVEN LIMITS  M      LIMITS
O SETS OF VALUES          $SET A LIST OF VARIABLES TO ONE OF TW M      CHOOSE
HINE ADDRESSES OF VARIABLES IN A LIST     $CREATE VECTOR OF MAC M      XLOCV
                       $SHUFFLE A LIST OF INTEGERS FROM 1 TO N   F      SHUFFL
                          $SET A LIST OF VECTORS TO ZERO         M      STZS
WRITE OUTPUT TAPE WITH NORMAL OR LITERAL FORMAT VECTOR        $ F      FMTOUT
TPUT COLUMN VECTORS BY NORMAL OR LITERAL FORMATS             $OU M      CVSOUT
                       $ACCESS TO LITERAL OR ORDINARY FORMAT     M      FNDFMT
INE VECTOR OUTPUT WITH NORMAL OR LITERAL FORMAT            $OFFL F      VECOUT
  $OUTPUT VARIABLES BY NORMAL OR LITERAL FORMAT                  M      VRSOUT
OUTPUT NAMED VECTOR BY NORMAL OR LITERAL FORMAT WITH SPACING  $ F      VOUT
UTPUT NAMED VECTORS BY NORMAL OR LITERAL FORMATS WITH SPACING$O M      VSOUT
 BY PROXY CALL STATEMENTS      $LOCATE AND OPERATE SUBROUTINES M      LOCATE
O COMMON                       $LOCATE ARGUMENT WITH RESPECT T F      IXCARG
   $MOVE A VECTOR TO A DIFFERENT LOCATION                        M      MOVE
NT                          $STORE LOCATION WITH INDEXABLE ARGUME M      LOC
                            $LOGICAL SHIFT FUNCTION            M      LSHFT
                       $COMPUTE A LOGICAL SUMCHECK             M      FAPSUM
          $SHIFT VECTOR ELEMENTS LOGICALLY LEFT OR RIGHT       M      SHFTR2
LS                        $PERFORM LONG DIVISION OF TWO POLYNOMIA F      POLYDV
 $SCALE, CONVERT FLTG. VECTOR TO MACHINE INTEGERS OR CONVERSELY M      FXDATA
                       $FLOAT ANY MACHINE LANGUAGE INTEGER     M      FLOATM
UIVALENT HOLLERITH       $CONVERT MACHINE LANGUAGE INTEGER TO EQ M      MLI2A6
 IN A LIST     $CREATE VECTOR OF MACHINE ADDRESSES OF VARIABLES M      XLOCV
 OR ROUND FLOATING PT. NUMBER TO MACHINE INTEGER      $TRUNCATE M      XFIXM
R      $FAST SQUARE ELEMENTS OF A MACHINE LANGUAGE INTEGER VECTO M      SQRMLI
R  $PRINT OR WPITE OUTPUT TAPE A MACHINE LANGUAGE INTEGER VECTO F      PWMLIV
IDAL INTEGRAL OF FUNCTION OR ITS MAGNITUDE    $DEFINITE TRAPEZO M      TINGL
 AN INTEGER SERIES             $MAP A SEQUENCE OF NUMBERS INTO M      MPSEQ1
RELATION OF TRANSIENT VECTORS OF MATRICES              $CROSSCOR F      CRSVM
VERSED DOT PRODUCT OF VECTORS OF MATRICES    $DOT PRODUCT OR RE F      MDOT3
VERSED DOT PRODUCT OF VECTORS OF MATRICES    $DOT PRODUCT OR RE F      MDOT
```

```
              $REVERSE VECTOR OF MATRICES                                F   MRVRS
OR A SYMMETRIC POSITIVE DEFINITE MATRIX                        $FACT F   MFACT
                    $INVERSE OF A MATRIX                               F   MATINV
KED OFF-LINE            $OUTPUT A MATRIX AS INTEGERS DENSELY PAC F   MOUTAI
ICATION                   $N X M MATRIX BY M X L MATRIX MULTIPL F   MATML3
NDEX BY INTERPOLATION    $FIND A MATRIX COLUMN WITH ARBITRARY I M   ARBCOL
              $EXTREMAL VALUES OF MATRIX ELEMENTS                  M   MAXSNM
          $N X M MATRIX BY M X L MATRIX MULTIPLICATION             F   MATML3
                        $SQUARE MATRIX MULTIPLICATION              M   MATML1
ELS                   $CONTOUR A MATRIX ON THE PRINTER IN DECIB F   CNTRDB
                              $MATRIX OUTPUT IN G FORMAT           F   MOUT
TER                   $CONTOUR OF MATRIX SUBSET ON OFF-LINE PRIN F   CONTUR
                              $MATRIX TRANSPOSE                    M   MATRA
                           $SQUARE MATRIX TRANSPOSE                M   MATRA1
TION FUNCTIONS         $REGION TO MAXIMIZE RATIO OF TWO DISTRIBU F   MXRARE
    $NORMALIZE A VECTOR TO GIVEN MAXIMUM VALUE                     M   NMZMG1
PENDENCY FROM PROBABILITY DENSIT$MEAN SQUARE CONTINGENCY AND DE F   MSCON1
                      $REMOVE THE MEAN FROM A FIXED VECTOR         M   XREMAV
                      $REMOVE THE MEAN FROM A FLOATING VECTOR      M   REMAV
           $NORMALIZE AND CHANGE MEAN OF A VECTOR                  F   NRMVEC
R                        $MOVING MEAN SQUARE AVERAGE OF A VECTO F   MVSQAV
MON             $OFF-LINE PRINT OF MEMORY USAGE - PROGRAM AND COM F   MEMUSE
SE ODD-LENGTHED VECTOR ABOUT ITS MIDPOINT                $COLLAP M   KOLAPS
F $ACCESS ROUTINE FOR RAND CORP. MILLION RANDOM DIGITS FROM TAP F   GETRD1
  $FACTOR POWER SPECTRUM TO FIND MINIMUM PHASE WAVELET             M   FACTOR
                           $ENABLE MIXED EXPRESSIONS IN FORTRAN    M   SAME
ONVERT FORTRAN INTEGER VECTOR TO MLI VECTOR                $FAST C M   ITOMLI
POINT INTEGER         $MULTIPLY AN MLI VECTOR BY A FORTRAN FIXED   M   MLISCL
      $QUICK CROSSCORRELATION OF MLI TRANSIENTS                    F   QXCOR1
     $FAST COMPARE TWO ARBITRARY MODE VECTORS FOR IDENTITY         M   CMPARV
 VECTOR EQUAL TO A CONSTANT (ANY MODE)     $SET ALL ELEMENTS OF M   SETKV
 CONSTANT OR BY CONSTANTS          $MODIFY A SET OF VARIABLES BY A M   ADDK
IONS FOR DANIELL SPECTRA           $MODIFY AUTO- OR CROSS-CORRELAT M   ADANL
LITY DISTRIBUTION WITH SPECIFIED MOMENTS      $GENERATE PROBABI F   PRBFIT
$REREAD DATA RECORD AND END FILE MONITOR                           M   REREAD
                $OPTIONAL ONLINE MONITOR OF BCD TAPE WRITING       M   ONLINE
ING BEHAVIOR   $CHECK VECTOR FOR MOMOTONE INCREASING OR DECREAS M   MONOCK
TERPOLATION      $INVERSION OF A MONOTONE FUNCTION BY LINEAR IN M   IFNCTN
OCATION                            $MOVE A VECTOR TO A DIFFERENT L M   MOVE
RS                                 $MOVE AN ARBITRARY SET OF VECTO M   MOVECS
OR CHANGE SIGN OF A VECTOR         $MOVE,REVERSE, CHANGE SPACING, M   MOVREV
                                   $MOVE DATA BLOCK                M   MVBLOK
                                   $MOVING AVERAGE OF A VECTOR     F   MVINAV
          $TRIANGULAR AVERAGING, MOVING LEFT OR RIGHT END          M   TAMVL
A VECTOR                           $MOVING MEAN SQUARE AVERAGE OF  F   MVSQAV
INT VECTOR                   $FAST MOVING SUMMATION OF A FIXED PO M   MUVADD
 BY A CONSTANT                     $MOVING SUMMATION WITH DIVISION M   MVNSUM
 ABSOLUTE VALUE INTEGRAL           $MOVING TRAPEZOIDAL INTEGRAL OR M   MVNTIN
UARES                              $MULTI-INPUT FILTER BY LEAST SQ F   MIFLS
 SQUARES                           $MULTI-INPUT PREDICTOR BY LEAST F   MIPLS
```

```
N                                   $MULTI-INPUT SIDEWARDS ITERATIO F     MISS
   $N X M MATRIX BY M X L MATRIX MULTIPLICATION                      F     MATML3
                  $SQUARE MATRIX MULTIPLICATION                      M     MATML1
RTRAN FIXED POINT INTEGER         $MULTIPLY AN MLI VECTOR BY A FO M     MLISCL
BY A SINGLE FLTG. PT. CONSTANT    $MULTIPLY ANY NO. OF VARIABLES  F     MULK  -II
RS FIXED OR FLOATING              $MULTIPLY ELEMENTS OF TWO VECTO M     VTIMSV
 FIXED CONSTANT                   $MULTIPLY VECTOR BY FLOATING OR M     MULPLY
ON INTO EQUALLY LIKELY SECTIONS   $NORMAL DISTRIBUTION AND DIVISI M     NOINT1
XIMUM VALUE                       $NORMALIZE A VECTOR TO GIVEN MA M     NMZMG1
 VECTOR                           $NORMALIZE AND CHANGE MEAN OF A F     NRMVEC
N GIVEN RANGE$FREQUENCY COUNT OF  NUMBER OF VALUES OF A SERIES I M     FRQCT2
 LAST TERM    $SEARCH VECTOR FOR  NUMBER, STARTING FROM FIRST OR F     SRCH1
 $TRUNCATE OR ROUND FLOATING PT. NUMBER TO MACHINE INTEGER       M     XFIXM
              $MAP A SEQUENCE OF NUMBERS INTO AN INTEGER SERIES M     MPSEQ1
ER FORTRAN-II INTEGER IS EVEN OR ODD          $DETERMINE WHETH M     XOOZE
SPLIT A VECTOR INTO ITS EVEN AND ODD PARTS (OR INVERSE)       $ M     SPLIT
COSINE AND/OR SINE TRANSFORMS OF ODD-LENGTH SERIES      $FAST   F     COSIS1
VIDPOINT                $COLLAPSE ODD-LENGTHED VECTOR ABOUT ITS  M     KOLAPS
SINE TRANSFORMS FROM 2 OR 4 EVEN-ODD PARTS $FAST COSINE AND/OR  M     COSP
MAL OR LITERAL FORMAT             $OFFLINE VECTOR OUTPUT WITH NOR F     VECOUT
ATRIX AS INTEGERS DENSELY PACKED OFF-LINE          $OUTPUT A M F     MOUTAI
 - PROGRAM AND COMMON             $OFF-LINE PRINT OF MEMORY USAGE F     MEMUSE
    $CONTOUR OF MATRIX SUBSET ON OFF-LINE PRINTER                 F     CONTUR
      $FAST COSINE TRANSFORMS OF ONE-SIDED AUTOCORRELATIONS       M     ASPECT
RANGE                   $COLLAPSE ONE-SIDED VECTOR INTO SMALLER  M     COLAPS
TING                    $OPTIONAL ONLINE MONITOR OF BCD TAPE WRI M     ONLINE
ALL STATEMENTS       $LOCATE AND OPERATE SUBROUTINES BY PROXY C M     LOCATE
CED DATA VALUES   $INTERPOLATION OPERATOR FOR 1 TO 4 EVENLY SPA M     INTOPR
 ONE SUBROUTINE REPEATEDLY        $OPERATE SEVERAL SUBROUTINES OR M     SEVRAL
INE TO GIVEN ACCURACY       $FIND OPERATION TIME OF NEXT SUBROUT M     TIMSUB
          $FAST REVERSE STORAGE ORDER OF A VECTOR                 M     REVERS
ORE SUBROUTINE DISPLA  $VARIABLE ORIGIN FORMAT GENERATOR FOR SC M     DSPFMT
ER TRANSFORM WITH ARBITRARY TIME ORIGIN    $QUICK INVERSE FOURI F     QIFURY
OF TRANSIENT WITH ARBITRARY TIME ORIGIN$FAST FOURIER TRANSFORM  F     QFURRY
ILIARY INFORMATION FOR AN INDATA-OUDATA TYPE TAPE     $LIST AUX F     LISTING
TIALIZED FOR ADDING TO AN INDATA-OUDATA TAPE                $INI F     SETINO
          $TERMINATE AN INDATA-OUDATA TAPE                      F     TRMINO
NSELY PACKED OFF-LINE             $OUTPUT A MATRIX AS INTEGERS DE F     MOUTAI
L OR LITERAL FORMATS              $OUTPUT COLUMN VECTORS BY NORMA M     CVSOUT
                          $MATRIX OUTPUT IN G FORMAT              F     MOUT
ERAL FORMAT VECTOR         $WRITE OUTPUT TAPE WITH NORMAL OR LIT F     FMTOUT
 IN G FORMAT                      $OUTPUT VARIABLES FIVE PER LINE M     CSOUT
OR LITERAL FORMAT WITH SPACING    $OUTPUT NAMED VECTOR BY NORMAL  F     VOUT
 OR LITERAL FORMATS WITH SPACING$OUTPUT NAMED VECTORS BY NORMAL M     VSOUT
 FORMAT OF A SUCCEEDING INPUT OR OUTPUT STATEMENT  $REPLACE THE M     RPLFMT
 INTEGER VECTOR  $PRINT OR WRITE OUTPUT TAPE A MACHINE LANGUAGE F     PWMLIV
LITERAL FORMAT                    $OUTPUT VARIABLES BY NORMAL OR  M     VRSOUT
FORMAT            $OFFLINE VECTOR OUTPUT WITH NORMAL OR LITERAL  F     VECOUT
 AS HOLLERITH VECTOR              $PACK UP FORTRAN INTEGER VECTOR M     IVTOHV
R    $SCALE AND FIX DATA VECTOR, PACK N DATA POINTS PER REGISTE M     PAKN
```

```
PUT A MATRIX AS INTEGERS DENSELY PACKED OFF-LINE                $OUT F      MOUTAI
        $UNPACK AND RESCALE A PACKED DATA VECTOR                    M      UNPAKN
PACE CARRIAGE N LINES OR RESTORE PAGE                          $S F      CARIGE
D OR FLOATING VECTOR ELEMENTS IN PAIRS          $DIFFERENCE FIXE M      DIFPRS
 VARIABLES FOR EQUALITY $COMPARE PAIRS OF VARIABLES OR A SET OF M      CMPARP
 TRANSFORMS FROM 2 OR 4 EVEN-ODD PARTS $FAST COSINE AND/OR SINE M      COSP
T A VECTOR INTO ITS EVEN AND ODD PARTS (OR INVERSE)       $SPLI M      SPLIT
 OR REVERSE       $AMPLITUDE AND PHASE FROM REAL AND IMAGINARY, M      AMPHZ
R POWER SPECTRUM TO FIND MINIMUM PHASE WAVELET           $FACTO M      FACTOR
COPE                              $PLOT FAST HORIZONTAL LINE ON S M 7090LINEH
COPE                              $PLOT FAST HORIZONTAL LINE ON S M  709LINEH
PE                                $PLOT FAST VERTICAL LINE ON SCO M 7090LINEV
PE                                $PLOT FAST VERTICAL LINE ON SCO M  709LINEV
VECTORS                     $PRINTER PLOT OF A SET OF EQUAL LENGTH  F      PLTVS1
RS                          $PRINTER-PLOT OF ARBITRARY SET OF VECTO F      PLOTVS
          $MULTIPLE FRAME SCOPE PLOTS OF VECTOR SETS                F      GRAPH
        $FIND CONTOUR LEVELS FOR PLOTTING A ROW OF DATA             F      CNTROW
                    $BAR GRAPH PLOTTING FOR SUBROUTINE GRAPH        M      HSTPLT-II
              $CUBIC CURVE SCOPE PLOTTING FOR SUBROUTINE GRAPH      M      HSTPLT-III
              $CUBIC CURVE SCOPE PLOTTING FOR SUBROUTINE GRAPH      M      HSTPLT-III
                    $HISTOGRAM PLOTTING FOR SUBROUTINE GRAPH        M      HSTPLT
                               $PLURALIZE THE NEXT SUBROUTINE       M      PLURNS
S SETK AND SETVEC              $PLURALIZED FORMS OF SUBROUTINE      M      SETKP
CH EXACTLY FITS 3 EQUALLY SPACED POINTS     $FIND QUADRATIC WHI M      QUFIT1
AND FIX DATA VECTOR, PACK N DATA POINTS PER REGISTER    $SCALE  M      PAKN
ER SEQUENCE IN GROUPS OF FIVE AS POKER HAND$EVALUATION OF INTEG F      POKCT1
                         $COMPLEX POLYNOMIAL EVALUATION             F      IPLYEV
HE POWER SERIES SQUARE ROOT OF A POLYNOMIAL                $FIND T F      PSQRT
                               $POLYNOMIAL ROOT FINDER              F      MULLER
 AND COMPLEX ROOTS             $POLYNOMIAL SYNTHESIS FROM REAL      F      POLYSN
S REAL AND COMPLEX ROOTS       $POLYNOMIAL SYNTHESIZED FROM IT      F      PLYSYN
NTS FOR REAL ARGUMENT$EVALUATE A POLYNOMIAL WITH REAL COEFFICIE     F      POLYEV
   $PERFORM LONG DIVISION OF TWO POLYNOMIALS                        F      POLYDV
            $FACTOR A SYMMETRIC POSITIVE DEFINITE MATRIX            F      MFACT
 PHASE WAVELET          $FACTOR POWER SPECTRUM TO FIND MINIMUM     M      FACTOR
   $RAISE VECTOR TO POWER OR SUM POWER OF DEVIATIONS FROM BASE      M      POWER
NS FROM BASE    $RAISE VECTOR TO POWER OR SUM POWER OF DEVIATIO M      POWER
POLYNOMIAL                $FIND THE POWER SERIES SQUARE ROOT OF A  F      PSQRT
                     $MULTI-INPUT PREDICTOR BY LEAST SQUARES       F      MIPLS
SON LEAST SQUARE ERROR FILTER OR PREDICTOR          $WIENER-LEVIN F      WLLSFP
NSION  $REALIZABLE LEAST SQUARES PREDICTOR BY RECURSION, 1-DIME F      RLSPR
NSIONS $REALIZABLE LEAST SQUARES PREDICTOR BY RECURSION, 2-DIME F      RLSPR2
M AND COMMON              $OFF-LINE PRINT OF MEMORY USAGE - PROGRA F      MEMUSE
ACHINE LANGUAGE INTEGER VECTOR  $PRINT OR WRITE OUTPUT TAPE A M F      PWMLIV
OUR OF MATRIX SUBSET ON OFF-LINE PRINTER                   $CONT F      CONTUR
G 3-DIGIT INTEGERS         $LABEL PRINTER COLUMNS WITH INCREASIN F      COLABL
        $CONTOUR A MATRIX ON THE PRINTER IN DECIBELS               F      CNTRDB
 LENGTH VECTORS                $PRINTER PLOT OF A SET OF EQUAL F      PLTVS1
OF VECTORS                     $PRINTER-PLOT OF ARBITRARY SET  F      PLOTVS
$COMPUTE CHI-SQUARE FOR CONSTANT PROBABILITY CASE                  F      CHISQR
```

```
 CONTINGENCY AND DEPENDENCY FROM PROBABILITY DENSIT$MEAN SQUARE F      MSCON1
 VARIATE EXCEEDS A VALUE        $PROBABILITY THAT A CHI-SQUARED F      KIINT1
 SERIES AT GIVEN LAG      $SECODN PROBABILITY DENSITY OF INTEGER F      PROB2
SPECIFIED MOMENTS       $GENERATE PROBABILITY DISTRIBUTION WITH  F      PRBFIT
 $DIVIDE THE X AXIS INTO EQUALLY PROBABLE RANGES                 F      GRUP2
S                    $DISPLACED DOT PRODUCT OF 2-DIMENSIONAL ARRAY F      DOTP
                        $FAST DOT PRODUCT OF TWO VECTORS         M      FDOT
     $DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATRICES F      MDOT3
     $DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATRICES F      MDOT
T OF VECTORS OF MATRICES     $DOT PRODUCT OR REVERSED DOT PRODUC F      MDOT3
T OF VECTORS OF MATRICES     $DOT PRODUCT OR REVERSED DOT PRODUC F      MDOT
NTS                       $VECTOR DOT PRODUCT WITH ARBITRARY INCREME M      DOTJ
VISION BY CONSTANT           $DOT PRODUCT OF TWO VECTORS WITH DI M      VDOTV
OCATE AND OPERATE SUBROUTINES BY PROXY CALL STATEMENTS       $L M      LOCATE
ABLE                          $QUADRATIC INTERPOLATION IN A T F      QINTR1
 EQUALLY SPACED POINTS      $FIND QUADRATIC WHICH EXACTLY FITS 3 M      QUFIT1
TRANSIENTS                    $QUICK CROSSCORRELATION OF ML I  F      QXCOR1
M WITH ARBITRARY TIME ORIGIN     $QUICK INVERSE FOURIER TRANSFOR F      QIFURY
OWER OF DEVIATIONS FROM BASE     $RAISE VECTOR TO POWER OR SUM P M      POWER
TS FROM TAPE $ACCESS ROUTINE FOR RAND CORP. MILLION RANDOM DIGI F      GETRD1
S ROUTINE FOR RAND CORP. MILLION RANDOM DIGITS FROM TAPE $ACCES F      GETRD1
SE ONE-SIDED VECTOR INTO SMALLER RANGE                  $COLLAP M      COLAPS
R OF VALUES OF A SERIES IN GIVEN RANGE$FREQUENCY COUNT OF NUMBE M      FRQCT2
IFIED ACCURACY, OF GIVEN PROGRAM RANGE       $REAL TIME, TO SPEC M  709TIMA2B
ONE VECTOR FROM ANOTHER WITH NEW RANGE AND INCREMENT   $CREATE  M      NURINC
THE X AXIS INTO EQUALLY PROBABLE RANGES                $DIVIDE  F      GRUP2
TIONS        $REGION TO MAXIMIZE RATIO OF TWO DISTRIBUTION FUNC F      MXRARE
T                             $READ DATA IN GENERALIZED FORMA F      RDATA
RY TAPE                       $READ EVERY N-TH WORD FROM BINA N      PACDAT
       $AMPLITUDE AND PHASE FROM REAL AND IMAGINARY, OR REVERSE M      AMPHZ
ING 7090 INTERVAL CLOCK      $FOR REAL TIME TIMING IN SECONDS US M 7090CLOCK1
      $POLYNOMIAL SYNTHESIS FROM REAL AND COMPLEX ROOTS          F      POLYSN
$POLYNOMIAL SYNTHESIZED FROM ITS REAL AND COMPLEX ROOTS          F      PLYSYN
OMIAL WITH REAL COEFFICIENTS FOR REAL ARGUMENT$EVALUATE A POLYN F      POLYEV
UMENT$EVALUATE A POLYNOMIAL WITH REAL COEFFICIENTS FOR REAL ARG F      POLYEV
CY, OF GIVEN PROGRAM RANGE       $REAL TIME, TO SPECIFIED ACCURA M  709TIMA2B
CTOR BY RECURSION, 2-DIMENSIONS $REALIZABLE LEAST SQUARES PREDI F      RLSPR2
CTOR BY RECURSION, 1-DIMENSION  $REALIZABLE LEAST SQUARES PREDI F      RLSPR
R BY RECURSION                  $REALIZABLE LEAST SQUARES SHAPE F      RLSSR
                     $REREAD DATA RECORD AND END FILE MONITOR    M      REREAD
SITION TAPE   $TEST IF NEXT TAPE RECORD IS END OF FILE AND REPO M      ZEFBCD
  $SKIP FORWARD OR BACKWARD OVER RECORDS ON TAPE                 M      RSKIP
      $TWO-DIMENSIONAL FILTER BY RECURSION                       F      FIRE2
ALIZABLE LEAST SQUARES SHAPER BY RECURSION                $RE  F      RLSSR
ZABLE LEAST SQUARES PREDICTOR BY RECURSION, 1-DIMENSION  $REALI F      RLSPR
ZABLE LEAST SQUARES PREDICTOR BY RECURSION, 2-DIMENSIONS $REALI F      RLSPR2
CTOR THROUGH A CONSTANT          $REFLECT A FIXED OR FLOATING VE M      REFLEC
O DISTRIBUTION FUNCTIONS         $REGION TO MAXIMIZE RATIO OF TW F      MXRARE
A VECTOR, PACK N DATA POINTS PER REGISTER    $SCALE AND FIX DAT M      PAKN
ECTOR                            $REMOVE THE MEAN FROM A FIXED V M      XREMAV
```

```
G VECTOR                          $REMOVE THE MEAN FROM A FLOATIN M      REMAV
DING INPUT OR OUTPUT STATEMENT  $REPLACE THE FORMAT OF A SUCCEE M      RPLFMT
CK          $LIST DATA DECK AND REPOSITION TAPE TO FRONT OF DE F      DADECK
T TAPE RECORD IS END OF FILE AND REPOSITION TAPE   $TEST IF NEX M      ZEFBCD
E MONITOR                         $REREAD DATA RECORD AND END FIL M      REREAD
ERVAL TIMER IS ON MAKING ON-LINE REQUEST IF NOT   $CHECK IF INT F      CLKON
                      $UNPACK AND RESCALE A PACKED DATA VECTOR   M      UNPAKN
ICAL FILTER WITH GIVEN AMPLITUDE RESPONSE     $GENERATE SYMMETR F      GNFLT1
      $SPACE CARRIAGE N LINES OR RESTORE PAGE                   F      CARIGE
AL TAPE     $FAST AND CONVENIENT RETRIEVAL OF DATA FROM A SPECI F      INDATA
E FIRST                           $RETURN N-TH ARGUMENT BEYOND TH M      NTHA
S PRODUCED BY SPLIT.        $FAST REVERSAL OF SPECIAL VECTORS ,A M      CHPRTS
HASE FROM REAL AND IMAGINARY, OR REVERSE       $AMPLITUDE AND P M      AMPHZ
ANGE SIGN OF A VECTOR       $MOVE,REVERSE, CHANGE SPACING, OR CH M      MOVREV
                                  $REVERSE VECTOR OF MATRICES     F      MRVRS
IN PLACE                          $REVERSE A VECTOR ELSEWHERE OR  M      REVER
TOR                         $FAST REVERSE STORAGE ORDER OF A VEC M      REVERS
S OF MATRICES     $DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTOR F      MDOT3
S OF MATRICES     $DOT PRODUCT OR REVERSED DOT PRODUCT OF VECTOR F      MDOT
       $HOLLERITH LEFT ADJUST OR RIGHT ADJUST FUNCTION          M      HLADJ
 ELEMENTS ARITHMETICALLY LEFT OR RIGHT             $SHIFT VECTOR M      SHFTR1
CTOR ELEMENTS LOGICALLY LEFT OR RIGHT                  $SHIFT V M      SHFTR2
NGULAR AVERAGING, MOVING LEFT OR RIGHT END                $TRIA M      TAMVL
SE OR FROM TRUE AVERAGE         $R.M.S. DEVIATION FROM GIVEN BA M      RMSDEV
                     $POLYNOMIAL ROOT FINDER                  F      MULLER
UNDING                    $SQUARE ROOT OF A FIXED VECTOR WITH RO M      XSQRUT
                          $SQUARE ROOT OF A FLOATING VECTOR      M      SQROOT
   $FIND THE POWER SERIES SQUARE ROOT OF A POLYNOMIAL            F      PSQRT
 SYNTHESIS FROM REAL AND COMPLEX ROOTS              $POLYNOMIAL F      POLYSN
ESIZED FROM ITS REAL AND COMPLEX ROOTS        $POLYNOMIAL SYNTH F      PLYSYN
NWARDS AN ARBITRARY AMOUNT      $ROTATE A VECTOR UPWARDS OR DOW M      ROTAT1
ISYMMETRIC 2-DIMENSIONAL ARRAY  $ROTATE CENTRO-SYMMETRIC OR ANT F      ROAR2
           $ROUND, ROUND UP, OR ROUND DOWN A FLOATING VECTOR   M      RNDV
ACHINE INTEGER      $TRUNCATE OR ROUND FLOATING PT. NUMBER TO M M      XFIXM
OR TO NEAREST FLTG. PT. INTEGER $ROUND FLTG. PT. NO. UP, DOWN,  M      RND
 A FLOATING VECTOR              $ROUND, ROUND UP, OR ROUND DOWN M      RNDV
TING VECTOR                $ROUND, ROUND UP, OR ROUND DOWN A FLOA M      RNDV
 FLOATING VECTOR WITH OR WITHOUT ROUNDING                $FIX A M      FIXV
WO FIXED VECTORS WITH OR WITHOUT ROUNDING $DIVIDE ELEMENTS OF T M      XVDVBV
UARE ROOT OF A FIXED VECTOR WITH ROUNDING                  $SQ M      XSQRUT
FXD PT DIVIDE WITH TRUNCATION OR ROUNDING TO FORTRAN-II INTEGE$ M      XDIV
ND CONTOUR LEVELS FOR PLOTTING A ROW OF DATA                $FI F      CNTROW
DEFINITE INTEGRAL BY TRAPEZOIDAL RULE                       $IN M      INTGRA
 MACHINE INTEGERS OR CONVERSELY $SCALE, CONVERT FLTG. VECTOR TO M      FXDATA
K N DATA POINTS PER REGISTER    $SCALE AND FIX DATA VECTOR, PAC M      PAKN
RAL AND/OR INTEGRATE $UNSCALE OR SCALE VECTOR FOR SIMPSON INTEG F      SMPSON
COPE, CLIPPING EXCESSIVE VALUES $SCALE VECTOR TO INTEGERS FOR S M      SCPSCL
OR GREATER THAN GIVEN VALUE$FAST SCAN VECTOR FOR ELEMENT EQUAL  M      FASCN1
 OF VALUES ALL ABOVE GIVEN LEVEL$SCAN VECTOR FOR POSSIBLE BLOCK F      NXALRM
          $ADVANCE FILM FRAME ON SCOPE                         M 7090FRAME
```

```
          $ADVANCE FILM FRAME ON SCOPE                                  M  709FRAME
RITRARY STRAIGHT LINE SEGMENT ON SCOPE                        $FAST AR M 7090LINE
RITRARY STRAIGHT LINE SEGMENT ON SCOPE                        $FAST AR M  709LINE
   $PLOT FAST HORIZONTAL LINE ON SCOPE                                  M 7090LINEH
   $PLOT FAST HORIZONTAL LINE ON SCOPE                                  M  709LINEH
     $PLOT FAST VERTICAL LINE ON SCOPE                                  M 7090LINEV
     $PLOT FAST VERTICAL LINE ON SCOPE                                  M  709LINEV
        $WRITE HOLLERITH TEXT ON SCOPE                                  M 7090DISPLA
        $WRITE HOLLERITH TEXT ON SCOPE                                  M  709DISPLA
               $MULTIPLE FRAME SCOPE PLOTS OF VECTOR SETS               F      GRAPH
GRAPH             $CUBIC CURVE SCOPE PLOTTING FOR SUBROUTINE            M      HSTPLT-III
GRAPH             $CUBIC CURVE SCOPE PLOTTING FOR SUBROUTINE            M      HSTPLT-III
ABLE ORIGIN FORMAT GENERATOR FOR SCOPE SUBROUTINE DISPLA  $VARI         M      DSPFMT
ES $SCALE VECTOR TO INTEGERS FOR SCOPE, CLIPPING EXCESSIVE VALU         M      SCPSCL
                                $SEARCH A VECTOR FOR A VALUE            M      SEARCH
TING FROM FIRST OR LAST TERM    $SEARCH VECTOR FOR NUMBER, STAR         F      SRCH1
INTEGER SERIES AT GIVEN LAG     $SECODN PROBABILITY DENSITY OF          F      PROB2
OCK     $FOR REAL TIME TIMING IN SECONDS USING 7090 INTERVAL CL         M 7090CLOCK1
AND DIVISION INTO EQUALLY LIKELY SECTIONS $NORMAL DISTRIBUTION          M      NOINT1
   $FAST ARBITRARY STRAIGHT LINE SEGMENT ON SCOPE                       M 7090LINE
   $FAST ARBITRARY STRAIGHT LINE SEGMENT ON SCOPE                       M  709LINE
OR FLTG VECTOR EQUAL TO A LINEAR SEGMENT                 $ SET FXD      M      SETLIN
      $TEST THE CONDITION OF ANY SENSE SWITCH                           M      SWITCH
$GET HOLLERITH DATA FROM CALLING SEQUENCE                               F      GETHOL
TEGER SERIES                 $MAP A SEQUENCE OF NUMBERS INTO AN IN      M      MPSEQ1
POKER HAND$EVALUATION OF INTEGER SEQUENCE IN GROUPS OF FIVE AS          F      POKCT1
 FORTRAN VARIABLE LENGTH CALLING SEQUENCES                  $ENABLE     M      VARARG
            $FAST FUNCTIONS FOR SEQUENTIAL SINES AND COSINES            M      SEQSAC
OR SINE TRANSFORMS OF ODD-LENGTH SERIES       $FAST COSINE AND/         F      COSIS1
UENCE OF NUMBERS INTO AN INTEGER SERIES                $MAP A SEQ       M      MPSEQ1
Y COUNT OF NUMBER OF VALUES OF A SERIES IN GIVEN RANGE$FREQUENC         M      FRQCT2
TIONS FOR LONG, LIMITED ACCURACY SERIES       $FAST AUTOCORRELA         F      QACORR
TIONS FOR LONG, LIMITED ACCURACY SERIES            $FAST CONVOLU        F      QCNVLV
TIONS FOR LONG, LIMITED ACCURACY SERIES     $FAST CROSS-CORRELA         F      QXCORR
N PROBABILITY DENSITY OF INTEGER SERIES AT GIVEN LAG     $SECOD         F      PROB2
     $FAST CORRELATIONS FOR LONG SERIES OF FIXED POINT INTEGERS          M      PROCOR
MIAL             $FIND THE POWER SERIES SQUARE ROOT OF A POLYNO         F      PSQRT
 OF TWO SETS OF VALUES          $SET A LIST OF VARIABLES TO ONE         M      CHOOSE
 OR BY CONSTANTS       $MODIFY A SET OF VARIABLES BY A CONSTANT         M      ADDK
$COMPARE PAIRS OF VARIABLES OR A SET OF VARIABLES FOR EQUALITY          M      CMPARP
            $MOVE AN ARBITRARY SET OF VECTORS                           M      MOVECS
                               $SET A LIST OF VECTORS TO ZERO           M      STZS
AL TO A CONSTANT (ANY MODE)     $SET ALL ELEMENTS OF VECTOR EQU         M      SETKV
 TO SEPARATE VALUES (FXD OR FLTG$SET ANY NO. OF VARIABLES EQUAL         F      SETKS -II
 TO A SINGLE VALUE (FXD OR FLTG)$SET ANY NO. OF VARIABLES EQUAL         F      SETK  -II
O SEPARATE VALUES (FXD OR FLTG) $SET ANY NO. OF VECTORS EQUAL T         M      SETKVS
O A LINEAR SEGMENT                $ SET FXD OR FLTG VECTOR EQUAL T      M      SETLIN
OR FLOATING                     $SET LINEAR VECTORS, FIXED AND/         M      SETLNS
            $PRINTER PLOT OF A SET OF EQUAL LENGTH VECTORS              F      PLTVS1
      $PRINTER-PLOT OF ARBITRARY SET OF VECTORS                         F      PLOTVS
```

```
VEN VALUES                          $SET VARIABLES OR VECTORS TO GI M      SETK
                              $FAST SET VECTOR TO ZERO              M      STZ
IPLE FRAME SCOPE PLOTS OF VECTOR SETS                        $MULT F      GRAPH
 LIST OF VARIABLES TO ONE OF TWO SETS OF VALUES            $SET A M      CHOOSE
$PLURALIZED FORMS OF SUBROUTINES SETK AND SETVEC                  M      SETKP
ED FORMS OF SUBROUTINES SETK AND SETVEC                $PLURALIZ M      SETKP
S FOR INCREMENTING, TESTING, AND SETTING      $HYBRID SUBPROGRAM M      INDEX
                    $LEAST SQUARES SHAPER BY SIDEWAYS ITERATION   F      LSSS1
        $REALIZABLE LEAST SQUARES SHAPER BY RECURSION             F      RLSSR
                           $LOGICAL SHIFT FUNCTION                M      LSHFT
ICALLY LEFT OR RIGHT              $SHIFT VECTOR ELEMENTS ARITHMET M      SHFTR1
Y LEFT OR RIGHT                   $SHIFT VECTOR ELEMENTS LOGICALL M      SHFTR2
M 1 TO N                          $SHUFFLE A LIST OF INTEGERS FRO F      SHUFFL
                      $MULTI-INPUT SIDEWARDS ITERATION            F      MISS
         $LEAST SQUARES SHAPER BY SIDEWAYS ITERATION              F      LSSS1
ENTS              $FORM A VECTOR BY SIFTING ANOTHER AT EVEN INCREM M      SIFT
                     $CHANGE ALL SIGN BITS OF A VECTOR            M      CHSIGN
VERSE, CHANGE SPACING, OR CHANGE SIGN OF A VECTOR       $MOVE,RE M      MOVREV
VARIABLES OR 0 IF SAME INCLUDING SIGN $SIGN OF DIFFERENCE OF 2   M      XACTEQ
LES OR 0 IF SAME INCLUDING SIGN $SIGN OF DIFFERENCE OF 2 VARIAB M      XACTEQ
LUES OF A VECTOR               $FIND SIGNED OR UNSIGNED EXTREMAL VA M      MAXSN
ATE $UNSCALE OR SCALE VECTOR FOR SIMPSON INTEGRAL AND/OR INTEGR F      SMPSON
ERMINANT EVALUATION $SOLUTION OF SIMULTANEOUS EQUATIONS AND DET M      SIMEQ
R FLOATING   $GENERATE COSINE OR SINE HALF-WAVE TABLES, FIXED O M      COSTBL
EN-ODD PARTS $FAST COSINE AND/OR SINE TRANSFORMS FROM 2 OR 4 EV M      COSP
SERIES       $FAST COSINE AND/OR SINE TRANSFORMS OF ODD-LENGTH  F      COSIS1
LATION FUNCTIONS   $FAST COSINE, SINE TRANSFORMS OF CROSS-CORRE F      XSPECT
  $FAST FUNCTIONS FOR SEQUENTIAL SINES AND COSINES                M      SEQSAC
 $FAST MAKE INDEX (BY INCREASING SIZE) OF ELEMENTS IN A VECTOR    M      SIZEUP
FILES ON TAPE                     $SKIP FORWARD OR BACKWARD OVER  M      FSKIP
RECORDS ON TAPE                   $SKIP FORWARD OR BACKWARD OVER  M      RSKIP
ORE PAGE                          $SPACE CARRIAGE N LINES OR REST F      CARIGE
 $FAST EVALUATE CUBIC FOR EVENLY SPACED ARGUMENTS                 M      FASCUB
ATION OPERATOR FOR 1 TO 4 EVENLY SPACED DATA VALUES   $INTERPOL M      INTOPR
BIC WHICH EXACTLY FITS 4 EQUALLY SPACED POINTS          $FIND CU M      CUFIT1
ECTOR      $MOVE,REVERSE, CHANGE SPACING, OR CHANGE SIGN OF A V M      MOVREV
ROSSCORRELATION OF 2-DIMENSIONAL SPATIAL ARRAYS      $SPATIAL C F      SPCOR2
DIMENSIONAL SPATIAL ARRAYS      $SPATIAL CROSSCORRELATION OF 2- F      SPCOR2
           $FAST TWO-DIMENSIONAL SPATIAL SPECTRUM                F      PLANSP
R CROSS-CORRELATIONS FOR DANIELL SPECTRA        $MODIFY AUTO- O M      ADANL
            $HIGH SPEED 24 POINT SPECTRUM                        F      FT24  -II
            $HIGH SPEED 24 POINT SPECTRUM                        M      FT24
 WAVELET           $FACTOR POWER SPECTRUM TO FIND MINIMUM PHASE M      FACTOR
   $FAST TWO-DIMENSIONAL SPATIAL SPECTRUM                        F      PLANSP
                          $HIGH SPEED 24 POINT SPECTRUM          F      FT24  -II
                          $HIGH SPEED 24 POINT SPECTRUM          M      FT24
DER CUBIC INTERPOLATION      $HI-SPEED EXPANSION OF A VECTOR UN M      EXPAND
 SPECIAL VECTORS ,AS PRODUCED BY SPLIT.       $FAST REVERSAL OF M      CHPRTS
ND ODD PARTS (OR INVERSE)         $SPLIT A VECTOR INTO ITS EVEN A M      SPLIT
 FORTRAN INTEGERS                 $SPREAD OUT HOLLERITH VECTOR AS M      HVTOIV
```

```
NCY FROM PROBABILITY DENSIT$MEAN SQUARE CONTINGENCY AND DEPENDE F     MSCON1
Y CASE                $COMPUTE CHI-SQUARE FOR CONSTANT PROBABILIT F     CHISQR
                                $SQUARE MATRIX MULTIPLICATION    M     MATML1
                                $SQUARE MATRIX TRANSPOSE         M     MATRA1
                      $MOVING MEAN SQUARE AVERAGE OF A VECTOR    F     MVSQAV
M ANOTHER OR FROM A CONSTANT$SUM SQUARE DIF. OF FLTG VECTOR FRO M     SQRDFR
M ANOTHER OR FROM A CONSTANT$SUM SQUARE DIF. OR FXD. VECTOR FRO M     XSQDFR
ANGUAGE INTEGER VECTOR     $FAST SQUARE ELEMENTS OF A MACHINE L M     SQRMLI
 VECTOR                        $SQUARE ELEMENTS OF FXD OR FLTG M     SQUARE
OR         $WIENER-LEVINSON LEAST SQUARE ERROR FILTER OR PREDICT F     WLLSFP
WITH ROUNDING                  $SQUARE ROOT OF A FIXED VECTOR  M     XSQRUT
OR                             $SQUARE ROOT OF A FLOATING VECT M     SQROOT
          $FIND THE POWER SERIES SQUARE ROOT OF A POLYNOMIAL    F     PSQRT
E        $PROBABILITY THAT A CHI-SQUARED VARIATE EXCEEDS A VALU F     KIINT1
FXD VECTOR               $SUM THE SQUARED ELEMENTS OF A FLTG OR  M     SQRSUM
    $MULTI-INPUT FILTER BY LEAST SQUARES                       F     MIFLS
 $MULTI-INPUT PREDICTOR BY LEAST SQUARES                       F     MIPLS
                          $LEAST SQUARES LINE                  F     LSLINE
RATION                    $LEAST SQUARES SHAPER BY SIDEWAYS ITE F     LSSS1
• 1-DIMENSIO .  $REALIZABLE LEAST SQUARES PREDICTOR BY RECURSION F     RLSPR
• 2-DIMENSIONS $REALIZABLE LEAST SQUARES PREDICTOR BY RECURSION F     RLSPR2
               $REALIZABLE LEAST SQUARES SHAPER BY RECURSION    F     RLSSR
 OF A SUCCEEDING INPUT OR OUTPUT STATEMENT  $REPLACE THE FORMAT M     RPLFMT
PERATE SUBROUTINES BY PROXY CALL STATEMENTS       $LOCATE AND O M     LOCATE
IXED POINT   $DELTA FUNCTION AND STEP FUNCTIONS, FLOATING AND F M     DELTA
          $FIND LENGTH OF COMMON STORAGE                       M     XLCOMN
         $FAST AND CONVIENT DATA STORAGE ON TAPE               F     OUDATA
                  $FAST REVERSE STORAGE ORDER OF A VECTOR      M     REVERS
EMENTS       $LOCATE AND OPERATE SUBROUTINES BY PROXY CALL STAT M     LOCATE
REPEATEDLY      $OPERATE SEVERAL SUBROUTINES OR ONE SUBROUTINE  M     SEVRAL
            $PLURALIZED FORMS OF SUBROUTINES SETK AND SETVEC    M     SETKP
 VECTORS                  $ADD OR SUBTRACT TWO FLOATING OR FIXED M     VPLUSV
ANOTHER OR FROM A CONSTANT      $SUM DIFFERENCE OF VECTOR FROM  M     SUMDFR
XED VECTOR                      $SUM ELEMENTS OF FLOATING OR FI M     SUM
ASE    $RAISE VECTOR TO POWER OR SUM POWER OF DEVIATIONS FROM B M     POWER
 FROM ANOTHER OR FROM A CONSTANT$SUM SQUARE DIF. OF FLTG VECTOR M     SQRDFR
 FROM ANOTHER OR FROM A CONSTANT$SUM SQUARE DIF. OR FXD. VECTOR M     XSQDFR
FLTG OR FXD VECTOR              $SUM THE SQUARED ELEMENTS OF A  M     SQRSUM
              $COMPUTE A LOGICAL SUMCHECK                       M     FAPSUM
ED VECTOR           $INTEGRATED SUMMATION OF A FLOATING OF FIX M     INTSUM
ING BLOCKS OF CONSTANT LENGTH   $SUMMATION OF VECTOR OVER ABUTT M     BLKSUM
TOR                  $FAST MOVING SUMMATION OF A FIXED POINT VEC M     MUVADD
ONSTANT                   $MOVING SUMMATION WITH DIVISION BY A C M     MVNSUM
$TEST THE CONDITION OF ANY SENSE SWITCH                         M     SWITCH
TRIX                     $FACTOR A SYMMETRIC POSITIVE DEFINITE MA F     MFACT
IMENSIONAL ARRAY  $ROTATE CENTRO-SYMMETRIC OR ANTISYMMETRIC 2-D F     ROAR2
AMPLITUDE RESPONSE      $GENERATE SYMMETRICAL FILTER WITH GIVEN  F     GNFLT1
X ROOTS                $POLYNOMIAL SYNTHESIS FROM REAL AND COMPLE F     POLYSN
COMPLEX ROOTS          $POLYNOMIAL SYNTHESIZED FROM ITS REAL AND  F     PLYSYN
      $LINEAR INTERPOLATION IN A TABLE                          F     LINTR1
```

$QUADRATIC INTERPOLATION IN A	TABLE	F	QINTR1
ENERATE COSINE OR SINE HALF-WAVE	TABLES, FIXED OR FLOATING $G	M	COSTBL
CORP. MILLION RANDOM DIGITS FROM	TAPE $ACCESS ROUTINE FOR RAND	F	GETRD1
RETRIEVAL OF DATA FROM A SPECIAL	TAPE $FAST AND CONVENIENT	F	INDATA
MATION FOR AN INDATA-OUDATA TYPE	TAPE $LIST AUXILIARY INFOR	F	LISTING
ORWARD OR BACKWARD OVER FILES ON	TAPE $SKIP F	M	FSKIP
$FAST COPY FILE FROM ONE	TAPE TO ANOTHER - VERSION 2	M	CPYFL2
$LIST DATA DECK AND REPOSITION	TAPE TO FRONT OF DECK	F	DADECK
RMAT VECTOR $WRITE OUTPUT	TAPE WITH NORMAL OR LITERAL FO	F	FMTOUT
AST AND CONVIENT DATA STORAGE ON	TAPE $F	F	OUDATA
D FOR ADDING TO AN INDATA-OUDATA	TAPE $INITIALIZE	F	SETINO
READ EVERY N-TH WORD FROM BINARY	TAPE 3	N	PACDAT
WARD OR BACKWARD OVER RECORDS ON	TAPE $SKIP FOR	M	RSKIP
$TERMINATE AN INDATA-OUDATA	TAPE	F	TRMINO
RD IS END OF FILE AND REPOSITION	TAPE $TEST IF NEXT TAPE RECO	M	ZEFBCD
$WRITE BINARY DATA ON	TAPE	M	WRTDAT
R VECTOR $PRINT OR WRITE OUTPUT	TAPE A MACHINE LANGUAGE INTEGE	F	PWMLIV
REPOSITION TAPE $TEST IF NEXT	TAPE RECORD IS END OF FILE AND	M	ZEFBCD
$OPTIONAL ONLINE MONITOR OF BCD	TAPE WRITING	M	ONLINE
E	$TERMINATE AN INDATA-OUDATA TAP	F	TRMINO
D OF FILE AND REPOSITION TAPE	$TEST IF NEXT TAPE RECORD IS EN	M	ZEFBCD
E SWITCH	$TEST THE CONDITION OF ANY SENS	M	SWITCH
ID SUBPROGRAMS FOR INCREMENTING,	TESTING, AND SETTING $HYBR	M	INDEX
$WRITE HOLLERITH	TEXT ON SCOPE	M	7090DISPLA
$WRITE HOLLERITH	TEXT ON SCOPE	M	709DISPLA
090 INTERVAL CLOCK $FOR REAL	TIME TIMING IN SECONDS USING 7	M	7090CLOCK1
EN ACCURACY $FIND OPERATION	TIME OF NEXT SUBROUTINE TO GIV	M	TIMSUB
FOURIER TRANSFORM WITH ARBITRARY	TIME ORIGIN $QUICK INVERSE	F	QIFURY
FORM OF TRANSIENT WITH ARBITRARY	TIME ORIGIN$FAST FOURIER TRANS	F	QFURRY
F GIVEN PROGRAM RANGE $REAL	TIME, TO SPECIFIED ACCURACY, O	M	709TIMA2B
UEST IF NOT $CHECK IF INTERVAL	TIMER IS ON MAKING ON-LINE REQ	F	CLKON
NTERVAL CLOCK $FOR REAL TIME	TIMING IN SECONDS USING 7090 I	M	7090CLOCK1
CES $FAST	TRACK THROUGH A VECTOR OF INDI	M	FASTRK
$AUTOSPECTRUM BY COSINE	TRANSFORM OF AUTOCORRELATION	M	ASPEC2
BITRARY TIME ORIGIN$FAST FOURIER	TRANSFORM OF TRANSIENT WITH AR	F	QFURRY
ORIGIN $QUICK INVERSE FOURIER	TRANSFORM WITH ARBITRARY TIME	F	QIFURY
D PARTS $FAST COSINE AND/OR SINE	TRANSFORMS FROM 2 OR 4 EVEN-OD	M	COSP
S $FAST COSINE AND/OR SINE	TRANSFORMS OF ODD-LENGTH SERIE	F	COSIS1
RRELATIONS $FAST COSINE	TRANSFORMS OF ONE-SIDED AUTOCO	M	ASPECT
N FUNCTIONS $FAST COSINE, SINE	TRANSFORMS OF CROSS-CORRELATIO	F	XSPECT
$CROSSCORRELATION OF	TRANSIENT VECTORS OF MATRICES	F	CRSVM
ORIGIN$FAST FOURIER TRANSFORM OF	TRANSIENT WITH ARBITRARY TIME	F	QFURRY
$COMPLETE CONVOLUTION OF TWO	TRANSIENTS	M	CONVLV-II
$COMPLETE CONVOLUTION OF TWO	TRANSIENTS	F	CONVLV
LAG $CROSSCORRELATION OF	TRANSIENTS BEGINNING WITH ANY	F	CROST
LAG $CROSSCORRELATION OF	TRANSIENTS BEGINNING WITH ZERO	F	CROSS
$QUICK CROSSCORRELATION OF MLT	TRANSIENTS	F	QXCOR1
$MATRIX	TRANSPOSE	M	MATRA
$SQUARE MATRIX	TRANSPOSE	M	MATRA1
$INVERSION OF	TRAPEZOIDAL INTEGRAL	M	IINTGR

$INDEFINITE INTEGRAL BY TRAPEZOIDAL RULE	M	INTGRA
ON OR ITS MAGNITUDE $DEFINITE TRAPEZOIDAL INTEGRAL OF FUNCTI	M	TINGL
TE VALUE INTEGRAL $MOVING TRAPEZOIDAL INTEGRAL OR ABSOLU	M	MVNTIN
FET OR RIGHT END $TRIANGULAR AVERAGING, MOVING L	M	TAMVL
NUMBER TO MACHINE INTEGER $TRUNCATE OR ROUND FLOATING PT.	M	XFIXM
RAN-II INTEGE$FXD PT DIVIDE WITH TRUNCATION OR ROUNDING TO FORT	M	XDIV
RSION $TWO-DIMENSIONAL FILTER BY RECU	F	FIRE2
UM $FAST TWO-DIMENSIONAL SPATIAL SPECTR	F	PLANSP
TA VECTOR $UNPACK AND RESCALE A PACKED DA	M	UNPAKN
MPSON INTEGRAL AND/OR INTEGRATE $UNSCALE OR SCALE VECTOR FOR SI	F	SMPSON
VECTOR $FIND SIGNED OR UNSIGNED EXTREMAL VALUES OF A	M	MAXSN
A CHI-SQUARED VARIATE EXCEEDS A VALUE $PROBABILITY THAT	F	KIINT1
$FAST ABSOLUTE VALUE OF A VECTOR	M	ABSVAL
MENT EQUAL OR GREATER THAN GIVEN VALUE$FAST SCAN VECTOR FOR ELE	M	FASCN1
MALIZE A VECTOR TO GIVEN MAXIMUM VALUE $NOR	M	NMZMG1
$SEARCH A VECTOR FOR A VALUE	M	SEARCH
. OF VARIABLES EQUAL TO A SINGLE VALUE (FXD OR FLTG)$SET ANY NO	F	SETK -II
TRAPEZOIDAL INTEGRAL OR ABSOLUTE VALUE INTEGRAL $MOVING	M	MVNTIN
OR FOR 2 TO 4 EVENLY SPACED DATA VALUES $INTERPOLATION OPERAT	M	INTOPR
VARIABLES TO ONE OF TWO SETS OF VALUES $SET A LIST OF	M	CHOOSE
NGE$FREQUENCY COUNT OF NUMBER OF VALUES OF A SERIES IN GIVEN RA	M	FRQCT2
FIND SIGNED OR UNSIGNED EXTREMAL VALUES OF A VECTOR $	M	MAXSN
$EXTREMAL VALUES OF MATRIX ELEMENTS	M	MAXSNM
GUMENT FALLS INSIDE TWO LIMITING VALUES $FIND IF AR	M	XLIMIT
RS FOR SCOPE, CLIPPING EXCESSIVE VALUES $SCALE VECTOR TO INTEGE	M	SCPSCL
ET VARIABLES OR VECTORS TO GIVEN VALUES $S	M	SETK
CAN VECTOR FOR POSSIBLE BLOCK OF VALUES ALL ABOVE GIVEN LEVEL$S	F	NXALRM
NO. OF VECTORS EQUAL TO SEPARATE VALUES (FXD OR FLTG) $SET ANY	M	SETKVS
. OF VARIABLES EQUAL TO SEPARATE VALUES (FXD OR FLTG$SET ANY NO	F	SETKS -II
TORS $ALLOWS VARIABLE DEPTH INDEXING OF VEC	M	GETX
OR FOR SCOPE SUBROUTINE DISPLA $VARIABLE ORIGIN FORMAT GENERAT	M	DSPFMT
CES $ENABLE FORTRAN VARIABLE LENGTH CALLING SEQUEN	M	VARARG
CONSTANTS $MODIFY A SET OF VARIABLES BY A CONSTANT OR BY	M	ADDK
. CONSTANT $MULTIPLY ANY NO. OF VARIABLES BY A SINGLE FLTG. PT	F	MULK -II
ORMAT $OUTPUT VARIABLES FIVE PER LINE IN G F	M	CSOUT
E PAIRS OF VARIABLES OR A SET OF VARIABLES FOR EQUALITY $COMPAR	M	CMPARP
N GIVEN LIMITS $CHECK THAT VARIABLES FROM LIST FALL WITHI	M	LIMITS
S FOR EQUALITY $COMPARE PAIRS OF VARIABLES OR A SET OF VARIABLE	M	CMPARP
F VALUES $SET A LIST OF VARIABLES TO ONE OF TWO SETS O	M	CHOOSE
ZERO $CHOOSE BETWEEN TWO VARIABLES BY A THIRD ONE BEING	M	WHICH
FORMAT $OUTPUT VARIABLES BY NORMAL OR LITERAL	M	VRSOUT
LUE (FXD OR FLTG)$SET ANY NO. OF VARIABLES EQUAL TO A SINGLE VA	F	SETK -II
LUES (FXD OR FLTG$SET ANY NO. OF VARIABLES EQUAL TO SEPARATE VA	F	SETKS -II
E VECTOR OF MACHINE ADDRESSES OF VARIABLES IN A LIST $CREAT	M	XLOCV
NG SIGN $SIGN OF DIFFERENCE OF 2 VARIABLES OR 0 IF SAME INCLUDI	M	XACTEQ
VALUES $SET VARIABLES OR VECTORS TO GIVEN	M	SETK
$PROBABILITY THAT A CHI-SQUARED VARIATE EXCEEDS A VALUE	F	KIINT1
ANT TO ELEMENTS OF A FXD OR FLTG VECTOR $ADD A CONST	M	BOOST
$CHANGE ALL SIGN BITS OF A VECTOR	M	CHSIGN
$FAST ABSOLUTE VALUE OF A VECTOR	M	ABSVAL

RT FORTRAN INTEGER VECTOR TO MLI VECTOR	$FAST CONVE	M	ITOMLI
$FIND AVERAGE OF FLOATING VECTOR		M	AVRAGE
OR UNSIGNED EXTREMAL VALUES OF A VECTOR	$FIND SIGNED	M	MAXSN
$FLOAT A VECTOR		M	FLOATV
CY DISTRIBUTION OF A FIXED POINT VECTOR	$FREQUEN	F	FRQCT1
SUMMATION OF A FLOATING OR FIXED VECTOR	$INTEGRATED	M	INTSUM
NGE SPACING, OR CHANGE SIGN OF A VECTOR	$MOVE,REVERSE, CHA	M	MOVREV
TRAN INTEGER VECTOR AS HOLLERITH VECTOR	$PACK UP FOR	M	IVTOHV
PE WITH NORMAL OR LITERAL FORMAT VECTOR	$WRITE OUTPUT TA	F	FMTOUT
$COLLAPSE ODD-LENGTHED VECTOR	ABOUT ITS MIDPOINT	M	KOLAPS
$SPREAD OUT HOLLERITH VECTOR	AS FORTRAN INTEGERS	M	HVTOIV
$PACK UP FORTRAN INTEGER VECTOR	AS HOLLERITH VECTOR	M	IVTOHV
$DIVIDE A FLOATING VECTOR	BY A CONSTANT	M	DIVIDE
T INTEGER $MULTIPLY AN MLI VECTOR	BY A FORTRAN FIXED POIN	M	MLISCL
ARY INCREMENTS $VECTOR	DOT PRODUCT WITH ARBITR	M	DOTJ
$DIFFERENCE FIXED OR FLOATING VECTOR	ELEMENTS IN PAIRS	M	DIFPRS
$FAST DOUBLING OR HALVING OF A VECTOR	(FIXED OR FLOATING)	M	DUBLX
FATER THAN GIVEN VALUE$FAST SCAN VECTOR	FOR ELEMENT EQUAL OR GR	M	FASCN1
OR DECREASING BEHAVIOR $CHECK VECTOR	FOR MOMOTONE INCREASING	M	MONOCK
$COLLAPSE ONE-SIDED VECTOR	INTO SMALLER RANGE	M	COLAPS
$DERIVATIVE OF A VECTOR	OF DIFFERENCING	M	DERIVA
$FAST TRACK THROUGH A VECTOR	OF INDICES	M	STRK
$REVERSE VECTOR	OF MATRICES	F	MRVRS
CONSTANT LENGTH $SUMMATION OF VECTOR	OVER ABUTTING BLOCKS OF	M	BLKSUM
$MULTIPLE FRAME SCOPE PLOTS OF VECTOR	SETS	F	GRAPH
$MOVE A VECTOR	TO A DIFFERENT LOCATION	M	MOVE
CONVERSELY $SCALE, CONVERT FLTG. VECTOR	TO MACHINE INTEGERS OR	M	FXDATA
$FAST CONVERT FORTRAN INTEGER VECTOR	TO MLI VECTOR	M	ITOMLI
ON $HI-SPEED EXPANSION OF A VECTOR	UNDER CUBIC INTERPOLATI	M	EXPAND
G $FIX A FLOATING VECTOR	WITH OR WITHOUT ROUNDIN	M	FIXV
NCREASING SIZE) OF ELEMENTS IN A VECTOR	$FAST MAKE INDEX (BY I	M	SIZEUP
OVING SUMMATION OF A FIXED POINT VECTOR	$FAST M	M	MUVADD
$FAST REVERSE STORAGE ORDER OF A VECTOR		M	REVERS
TS OF A MACHINE LANGUAGE INTEGER VECTOR	$FAST SQUARE ELEMEN	M	SQRMLI
$FIND AVERAGE OF FIXED PT VECTOR		M	XAVRGE
$MOVING AVERAGE OF A VECTOR		F	MVINAV
$MOVING MEAN SQUARE AVERAGE OF A VECTOR		F	MVSQAV
$NORMALIZE AND CHANGE MEAN OF A VECTOR		F	NRMVEC
TAPE A MACHINE LANGUAGE INTEGER VECTOR	$PRINT OR WRITE OUTPUT	F	PWMLIV
$REMOVE THE MEAN FROM A FIXED VECTOR		M	XREMAV
$REMOVE THE MEAN FROM A FLOATING VECTOR		M	REMAV
UND UP, OR ROUND DOWN A FLOATING VECTOR	$ROUND, RO	M	RNDV
$SQUARE ELEMENTS OF FXD OR FLTG VECTOR		M	SQUARE
$SQUARE ROOT OF A FLOATING VECTOR		M	SQROOT
UM ELEMENTS OF FLOATING OR FIXED VECTOR	$S	M	SUM
QUARED ELEMENTS OF A FLTG OR FXD VECTOR	$SUM THE S	M	SQRSUM
UNPACK AND RESCALE A PACKED DATA VECTOR	$	M	UNPAKN
$DIVIDE A FXD VECTOR	BY A CONSTANT	M	XDVIDE
NSTANT $MULTIPLY VECTOR	BY FLOATING OR FIXED CO	M	MULPLY
RMAT WITH SPACING $OUTPUT NAMED VECTOR	BY NORMAL OR LITERAL FO	F	VOUT

```
VEN INCREMENTS              $FORM A VECTOR BY SIFTING ANOTHER AT E M     SIFT
         $DIVIDE ELEMENTS OF ONE VECTOR BY THOSE OF ANOTHER      M     VDVBYV
 LEFT OR RIGHT               $SHIFT VECTOR ELEMENTS ARITHMETICALLY M     SHFTR1
 OR RIGHT                    $SHIFT VECTOR ELEMENTS LOGICALLY LEFT M     SHFTR2
                        $REVERSE A VECTOR ELSEWHERE OR IN PLACE    M     REVER
Y MODE)      $SET ALL ELEMENTS OF VECTOR EQUAL TO A CONSTANT (AN M     SETKV
NT             $ SET FXD OR FLTG VECTOR EQUAL TO A LINEAR SEGME M     SETLIN
                      $SEARCH A VECTOR FOR A VALUE                 M     SEARCH
OM FIRST OR LAST TERM     $SEARCH VECTOR FOR NUMBER, STARTING FR F     SRCH1
ALUES ALL ABOVE GIVEN LEVEL$SCAN VECTOR FOR POSSIBLE BLOCK OF V F     NXALRM
D/OR INTEGRATE $UNSCALE OR SCALE VECTOR FOR SIMPSON INTEGRAL AN F     SMPSON
CONSTANT$SUM SQUARE DIF. OR FXD. VECTOR FROM ANOTHER OR FROM A  M     XSQDFR
CONSTANT       $SUM DIFFERENCE OF VECTOR FROM ANOTHER OR FROM A M     SUMDFR
CONSTANT$SUM SQUARE DIF. OF FLTG VECTOR FROM ANOTHER OR FROM A  M     SQRDFR
ANGE AND INCREMENT    $CREATE ONE VECTOR FROM ANOTHER WITH NEW R M     NURINC
ARTS (OR INVERSE)          $SPLIT A VECTOR INTO ITS EVEN AND ODD P M     SPLIT
 VARIABLES IN A LIST       $CREATE VECTOR OF MACHINE ADDRESSES OF M     XLOCV
ITERAL FORMAT              $OFFLINE VECTOR OUTPUT WITH NORMAL OR L F     VECOUT
 REGISTER     $SCALE AND FIX DATA VECTOR, PACK N DATA POINTS PER M     PAKN
    $REFLECT A FIXED OR FLOATING VECTOR THROUGH A CONSTANT       M     REFLEC
                       $NORMALIZE A VECTOR TO GIVEN MAXIMUM VALUE  M     NMZMG1
CLIPPING EXCESSIVE VALUES $SCALE VECTOR TO INTEGERS FOR SCOPE,  M     SCPSCL
F DEVIATIONS FROM BASE     $RAISE VECTOR TO POWER OR SUM POWER O M     POWER
                         $FAST SET VECTOR TO ZERO                  M     STZ
 ARBITRARY AMOUNT        $ROTATE A VECTOR UPWARDS OR DOWNWARDS AN M     ROTAT1
         $SQUARE ROOT OF A FIXED VECTOR WITH ROUNDING             M     XSORUT
LLOWS VARIABLE DEPTH INDEXING OF VECTORS                    $A M     GETX
               $EXCHANGE ANY TWO VECTORS                        M     EXCHVS
          $FAST DOT PRODUCT OF TWO VECTORS                      M     FDOT
        $MOVE AN ARBITRARY SET OF VECTORS                       M     MOVECS
        $FAST REVERSAL OF SPECIAL VECTORS ,AS PRODUCED BY SPLIT. M     CHPRTS
ORMATS              $OUTPUT COLUMN VECTORS BY NORMAL OR LITERAL F M     CVSOUT
$FAST COMPARE TWO ARBITRARY MODE VECTORS FOR IDENTITY            M     CMPARV
  $CROSSCORRELATION OF TRANSIENT VECTORS OF MATRICES             F     CRSVM
ODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATRICES    $DOT PR F     MDOT3
ODUCT OR REVERSED DOT PRODUCT OF VECTORS OF MATRICES    $DOT PR F     MDOT
R SUBTRACT TWO FLOATING OR FIXED VECTORS                 $ADD O M     VPLUSV
ER PLOT OF A SET OF EQUAL LENGTH VECTORS                 $PRINT F     PLTVS1
PRINTER-PLOT OF ARBITRARY SET OF VECTORS                       $ F     PLOTVS
ORMATS WITH SPACING$OUTPUT NAMED VECTORS BY NORMAL OR LITERAL F M     VSOUT
ES (FXD OR FLTG) $SET ANY NO. OF VECTORS EQUAL TO SEPARATE VALU M     SETKVS
                      $SET LINEAR VECTORS, FIXED AND/OR FLOAT+NG M     SETLNS
        $MULTIPLY ELEMENTS OF TWO VECTORS FIXED OR FLOATING      M     VTIMSV
               $SET VARIABLES OR VECTORS TO GIVEN VALUES         M     SETK
                  $SET A LIST OF VECTORS TO ZERO                 M     STZS
ANT          $DOT PRODUCT OF TWO VECTORS WITH DIVISION BY CONST M     VDOTV
NG $DIVIDE ELEMENTS OF TWO FIXED VECTORS WITH OR WITHOUT ROUNDI M     XVDVBV
   $GENERATE COSINE OR SINE HALF-WAVE TABLES, FIXED OR FLOATING M     COSTBL
R SPECTRUM TO FIND MINIMUM PHASE WAVELET          $FACTOR POWE M     FACTOR
                                $WIENER AUTOCORRELATION          F     WAC
```

RROR FILTER OR PREDICTOR	$WIENER-LEVINSON LEAST SQUARE E	F	WLLSFP
$READ EVERY N-TH WORD FROM BINARY TAPE		N	PACDAT
$COMPARE ARITHMETICALLY TWO WORDS WHERE -0 IS LESS THAN +0		M	CMPRA
	$WRITE HOLLERITH TEXT ON SCOPE	M	7090DISPLA
	$WRITE HOLLERITH TEXT ON SCOPE	M	709DISPLA
OR LITERAL FORMAT VECTOR	$WRITE OUTPUT TAPE WITH NORMAL	F	FMTOUT
	$WRITE BINARY DATA ON TAPE	M	WRTDAT
NGUAGE INTEGER VECTOR $PRINT OR WRITE OUTPUT TAPE A MACHINE LA		F	PWMLIV
IONAL ONLINE MONITOR OF BCD TAPE WRITING	$OPT	M	ONLINE
O VARIABLES BY A THIRD ONE BEING ZERO	$CHOOSE BETWEEN TW	M	WHICH
$FAST SET VECTOR TO ZERO		M	STZ
$SET A LIST OF VECTORS TO ZERO		M	STZS

7. Difference Between Programs Sets I and II

Additions 172 programs have been added to Set I in forming Set II. They are

ADDK	FASCUB	MDOT	QUFIT1	STZS
ARBCOL	FASTRK	MDOT3	QXCOR1	SUM
ARCTAN	FIRE2	MEMUSE	RDATA	SUMDFR
ASPEC2	FIXV	MFACT	REFLEC	SWITCH
AVRAGE	FLOATV	MIFLS	REMAV	TAMVL
BLKSUM	FMTOUT	MIPLS	REREAD	TIMA2B(7094)
BOOST	FNDFMT	MISS	REVER	TIMSUB
CARIGE	FT24-II	MONOCK	RLSPR	TINGL
CHOOSE	GETHOL	MOUT	RLSPR2	TRMINO
CHSIGN	GETX	MOUTAI	RLSSR	VDOTV
CLKON	GNHOL2	MOVECS	RMSDEV	VDVBYV
CMPARP	GRAPHX	MOVREV	RNDV	VECOUT
CMPARV	HLADJ	MRVRS	ROAR2	VOUT
CMPRA	HVTOIV	MULK-II	RPLFMT	VPLUSV
CNTRDB	IDERIV	MULLER	SEQSAC	VRSOUT
CNTROW	IFNCTN	MULPLY	SETINO	VSOUT
COLABL	IINTGR	MVINAV	SETK	VTIMSV
CONTUR	INDEX	MVNSUM	SETK-II	WHICH
COSIS1	INTGRA	MVNTIN	SETKP	WRTDAT
CPYFL2	INTHOL	MVSQAV	SETKS-II	XACTEQ
CROSS	INTOPR	MXRARE	SETKV	XAVRGE
CROST	INTSUM	NRMVEC	SETKVS	XDIV
CRSVM	IVTOHV	NTHA	SETLIN	XDVIDE
CSOUT	IXCARG	NURINC	SETLNS	XLCOMN
CUFIT1	LIMITS	ONLINE	SEVRAL	XLIMIT
CVSOUT	LOCATE	PACDAT	SHUFFL	XLOCV
DADECK	LSHFT	PLANSP	SIFT	XOOZE
DELTA	LSLINE	PLOTVS	SIZEUP	XREMAV
DERIVA	LSSS1	PLTVS1	SMPSON	XSQDFR
DIFFRS	MATINV	PLURNS	SPCOR2	XSQRUT
DIVIDE	MATML1	POLYSN	SQRDFR	XVDVBV
DOTJ	MATML3	POWER	SQROOT	ZEFBCD
DOTP	MATRA	QFURRY	SQRSUM	
EXCHVS	MATRA1	QIFURY	SQUARE	
EXPAND	MAXSNM	QINTR1	SRCH1	

Deletions 11 programs have been deleted from Set I in forming Set II. They are

ATSH	CRST1	GNFMT1	UPDATE
BENIMP	GETREC	ORGDLT	WRTREC
BENSPT	GETREC-II	ROKWIC	

Carryovers 95 programs were carried over from Set I to Set II. In all cases the date appearing on the first card of the symbolic deck has been changed and in most cases other changes have also been made, mostly to upgrade the documentation but in some cases to improve the coding. The carryovers are

ABSVAL	FSKIP	LOC	QXCORR
ADANL	FT24	MAXSN	REVERS
AMPHZ	FXDATA	MLISCL	RND
ASPECT	GENIOL	MLI2A6	ROTAT1
CHISQR	GETRD1	MOVE	RSKIP
CHPRTS	GNFLT1	MPSEQ1	SAME
CLOCK1(7090)	GRAPH	MSCON1	SCPSCL
COLAPS	GRUP2	MUVADD	SEARCH
CONVLV	HSTPLT	MVBLOK	SHFTR1
CONVLV-II	HSTPLT-II	NMZMG1	SHFTR2
COSP	HSTPLT-III(709)	NOINT1	SIMEQ
COSTBL	HSTPLT-III(7090)	NXALRM	SPLIT
DISPLA(709)	INDATA	OUDATA	SQRMLI
DISPLA(7090)	IPLYEV	PAKN	STZ
DSPFMT	ITOMLI	PLYSYN	UNPAKN
DUBLX	KIINT1	POKCT1	VARARG
FACTOR	KOLAPS	POLYDV	WAC
FAPSUM	LINE (709)	POLYEV	WLLSFP
FASCN1	LINE (7090)	PRBFIT	XFIXM
FDOT	LINEH (709)	PROB2	XSPECT
FLOATM	LINEH (7090)	PROCOR	
FRAME (709)	LINEV (709)	PSQRT	
FRAME (7090)	LINEV (7090)	PWMLIV	
FRQCT1	LINTR1	QACORR	
FRQCT2	LISTNG	QCNVLV	